FREE DVD

ADMIN Network & Security

OSDBuilder

Terraform

ADMIN

Network & Security

ISSUE 69

Terraform

Multicloud orchestrator milestone

PowerShell OSDBuilder

Maintain system images while enforcing security

KUBERNETES FOR SMEs

Windows Server 2022

Security features

MONITORING IN GCP

5 free wiki platforms reviewed

5G Campus Networks

High speed and low latency

grommunio

Microsoft Exchange replacement

PXE

Boot Server

Lithnet

Password protection for Active Directory

Cinc

Free and compatible Chef distribution

LINUX NEW MEDIA
The Pulse of Open Source

Do Something

Today, I just have a special message for you: Do something.

I know it's much easier to do nothing. In fact, doing nothing is always the default action, but doing something is better than doing nothing. I'm no mathematician, but I'd say it's about 100 percent better. Yes, there's a story here, and it's one you should read and heed lest you grow old without realizing your dreams.

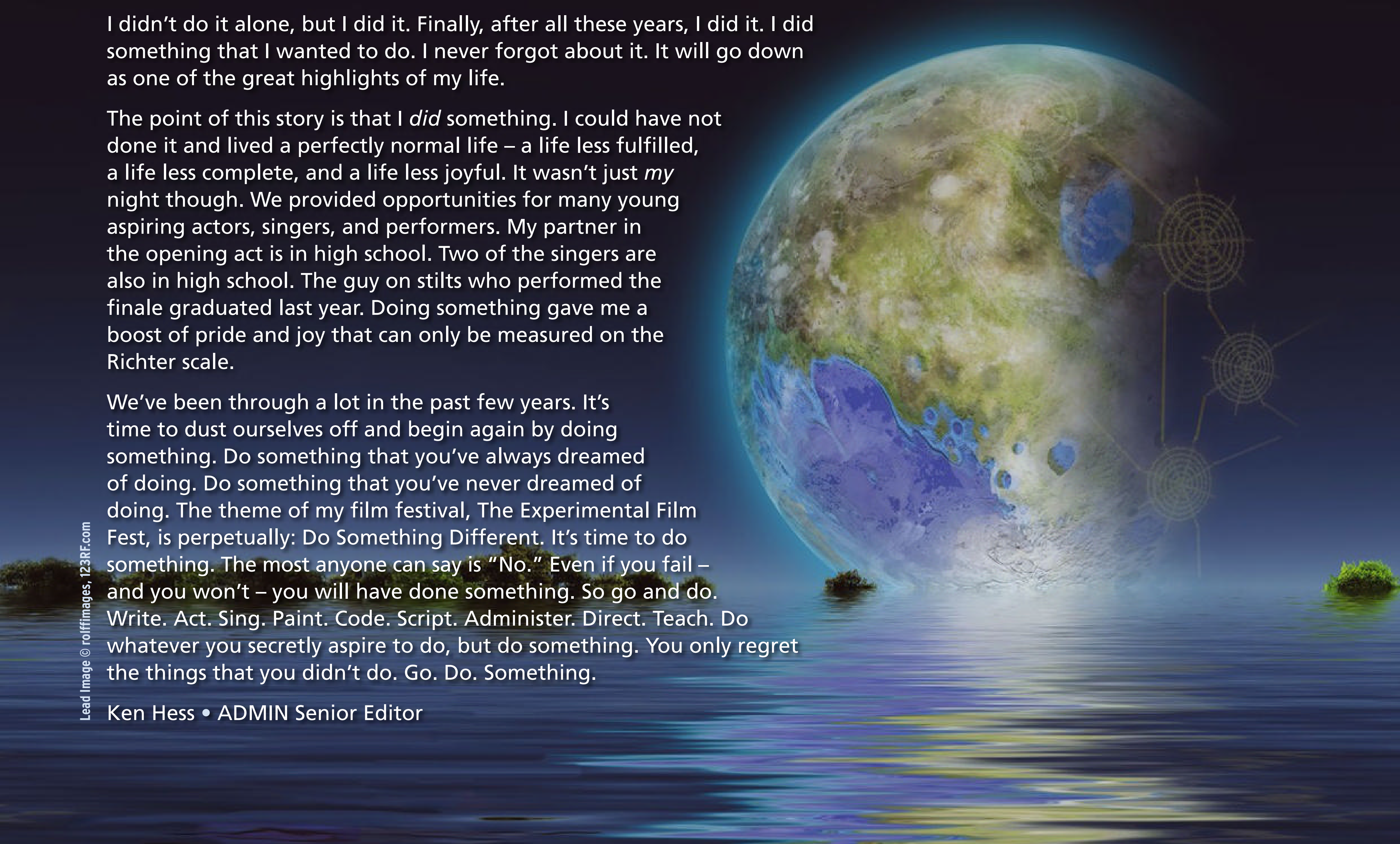
It all started when I went to college with dinosaurs and cave people in prehistoric times, aka the 1980s. Reagan was President. The recession was in full swing. Jobs were scarce. I wanted to produce and direct a vaudeville show. Yes, you read that correctly – a vaudeville show. Plate spinners, baton twirlers, bodacious burlesque acts, dog and pony shows, singers, dancers, comedy, magic, and a whole array of sideshow delights. Vaudeville. My dream to be a vaudevillian would never come to pass.

That is until many years later in 2022. I finally convinced someone to take a trip down memory lane with me back to those innocent days between the great wars and before the Great Depression. The heyday of vaudeville, the ultimate variety show. I found a theater director whose venue was on the original vaudeville circuit before it was stripped of its once great elegance and converted into a moving picture theater. I wanted to do it. I pitched it. And the vaudeville show was on! The date for the one-night-only show was April 2, 2022, in beautiful New Bern, North Carolina – my new home.

The show sold out and people loved it. Many have requested another show. "Vaudeville is here to stay," I stated bravely in the opening act. And I was right. It's now going to be part of the regular season and an annual event.

It turned out better than I'd hoped. We had a fire-breather, a flame-fan dancer, a guy on stilts, and an announcer who wrestled with a giant boa constrictor in the street in front of the theater before the show commenced inside. We had singers; comedians; tap dancers; side-show performances that included a bed of nails; glass walking; a dog act; bathtub acrobatics; an aerial silks performer; a belly dancer who did a sword dance; a tribute to the late, great Andy Kaufman; and much more. The first show of its kind in 100 years in that little theater in an obscure corner of eastern North Carolina.

I did it.



Lead Image © rolffimages, 123RF.com

I didn't do it alone, but I did it. Finally, after all these years, I did it. I did something that I wanted to do. I never forgot about it. It will go down as one of the great highlights of my life.

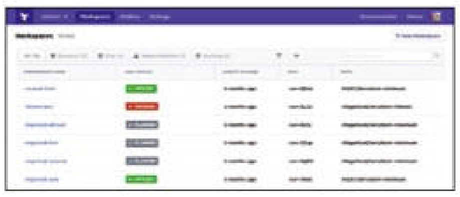
The point of this story is that I *did* something. I could have not done it and lived a perfectly normal life – a life less fulfilled, a life less complete, and a life less joyful. It wasn't just *my* night though. We provided opportunities for many young aspiring actors, singers, and performers. My partner in the opening act is in high school. Two of the singers are also in high school. The guy on stilts who performed the finale graduated last year. Doing something gave me a boost of pride and joy that can only be measured on the Richter scale.

We've been through a lot in the past few years. It's time to dust ourselves off and begin again by doing something. Do something that you've always dreamed of doing. Do something that you've never dreamed of doing. The theme of my film festival, The Experimental Film Fest, is perpetually: Do Something Different. It's time to do something. The most anyone can say is "No." Even if you fail – and you won't – you will have done something. So go and do. Write. Act. Sing. Paint. Code. Script. Administer. Direct. Teach. Do whatever you secretly aspire to do, but do something. You only regret the things that you didn't do. Go. Do. Something.

Ken Hess • ADMIN Senior Editor

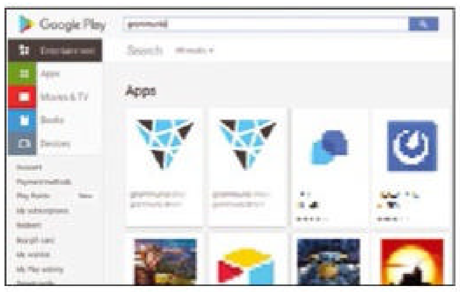
Features

After nearly 10 years of work on Terraform, the HashiCorp team delivers the 1.0 version of the cloud automation tool.



Tools

Save time and simplify your workday with these useful tools for real-world systems administration.



News

Find out about the latest ploys and toys in the world of information technology.

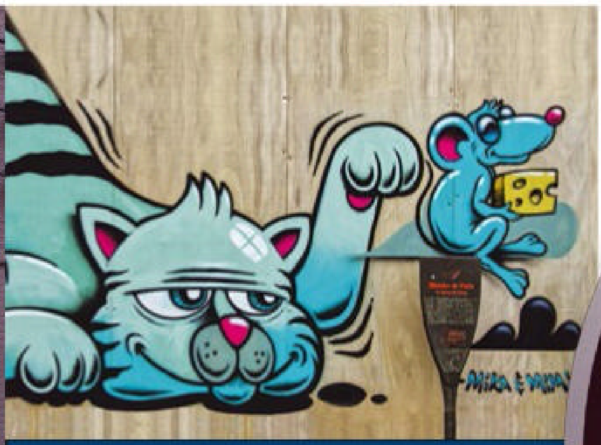
Tools

Containers and Virtualization

Virtual environments are becoming faster, more secure, and easier to set up and use. Check out these tools.

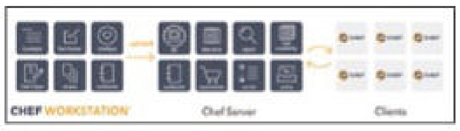
Security

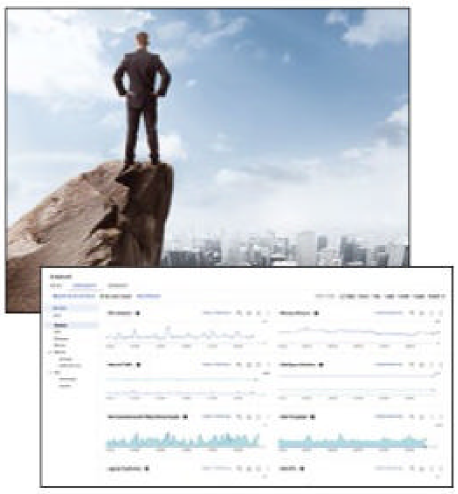
Use these powerful security tools to protect your network and keep intruders in the cold.



Management

Use these practical apps to extend, simplify, and automate routine admin tasks.





Nuts and Bolts

Timely tutorials on fundamental techniques for systems administrators.

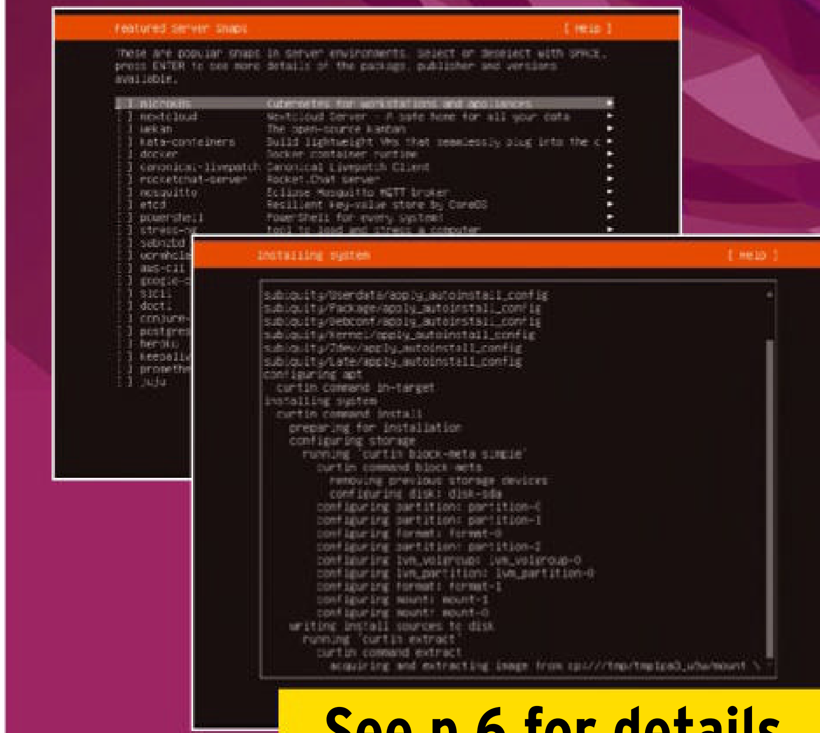
Service

While this ADMIN magazine disc has been tested and is to the best of our knowledge free of malicious software and defects, Admin Magazine cannot be held responsible and is not liable for any disruption, loss, or damage to data and computer systems related to the use of this disc.

ubuntu 22.04

"Jammy Jellyfish" LTS Server Edition

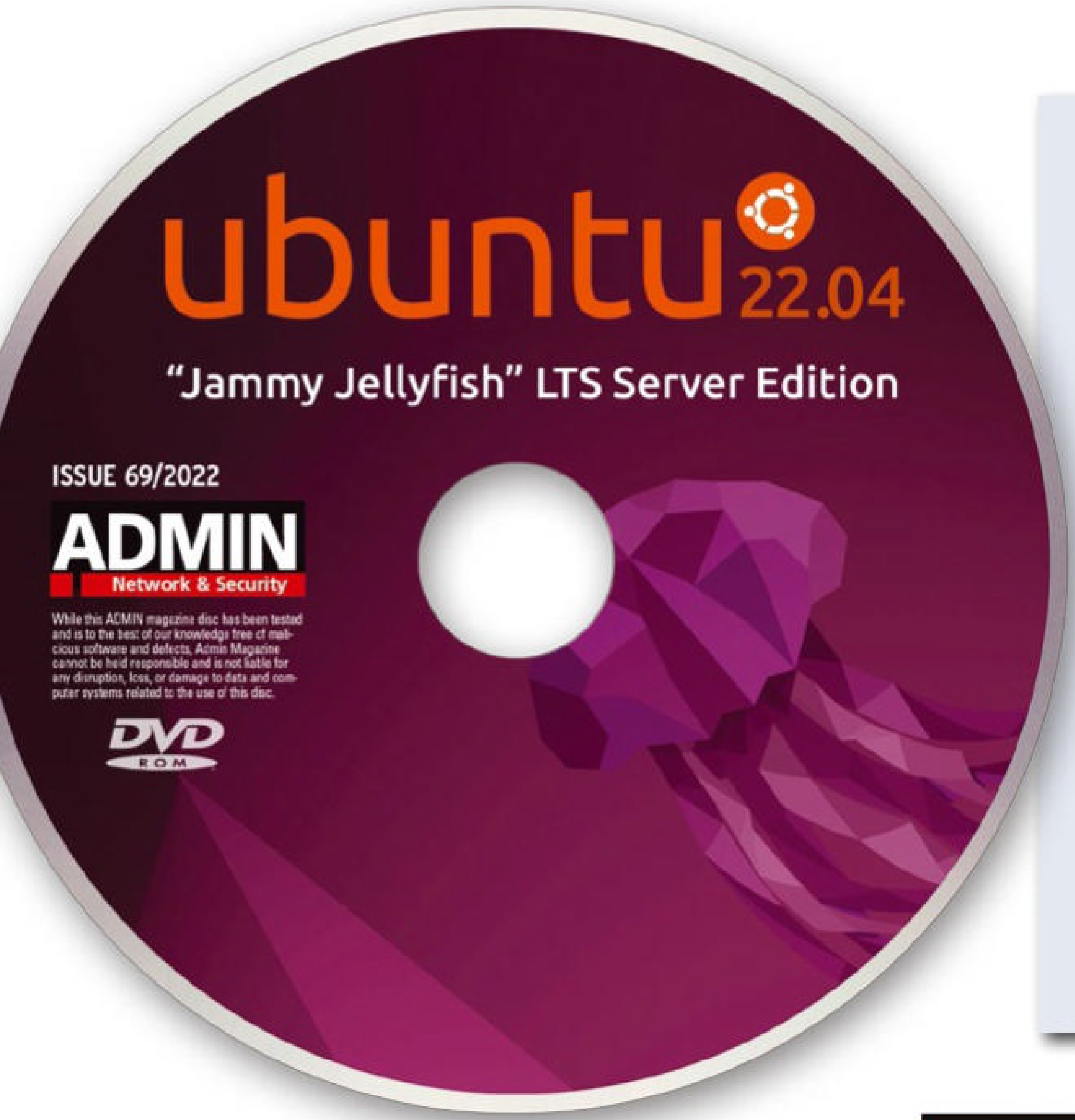
- OpenSSL 3.0
- Linux 5.15 kernel
- SmartNIC support in Netplan



```
Featured Server Snaps                                              [ Help ]

These are popular snaps in server environments. Select or deselect with SPACE,
press ENTER to see more details of the package, publisher and versions
available.

[ ] microk8s           Kubernetes for workstations and appliances          ▸
[ ] nextcloud          Nextcloud Server - A safe home for all your data    ▸
[ ] wekan              The open-source kanban                              ▸
[ ] kata-containers    Build lightweight VMs that seamlessly plug into the c ▸
[ ] docker             Docker container runtime                            ▸
[ ] canonical-livepatch Canonical Livepatch Client                         ▸
[ ] rocketchat-server  Rocket.Chat server                                  ▸
[ ] mosquitto          Eclipse Mosquitto MQTT broker                       ▸
[ ] etcd               Resilient key-value store by CoreOS                 ▸
[ ] powershell         PowerShell for every system!                        ▸
[ ] stress-ng          tool to load and stress a computer                  ▸
[ ] sabnzbd
[ ] wormhole
[ ] aws-cli
[ ] google-
[ ] slcli
[ ] doctl
[ ] conjure-
[ ] postgres
[ ] heroku
[ ] keepaliv
[ ] promethe
[ ] juju
```

```
Installing system                                                  [ Help ]

subiquity/Userdata/apply_autoinstall_config
subiquity/Package/apply_autoinstall_config
subiquity/Debconf/apply_autoinstall_config
subiquity/Kernel/apply_autoinstall_config
subiquity/Zdev/apply_autoinstall_config
subiquity/Late/apply_autoinstall_config
configuring apt
  curtin command in-target
installing system
  curtin command install
    preparing for installation
    configuring storage
      running 'curtin block-meta simple'
        curtin command block-meta
          removing previous storage devices
          configuring disk: disk-sda
        configuring partition: partition-0
        configuring partition: partition-1
        configuring format: format-0
        configuring partition: partition-2
        configuring lvm_volgroup: lvm_volgroup-0
        configuring lvm_partition: lvm_partition-0
        configuring format: format-1
        configuring mount: mount-1
        configuring mount: mount-0
    writing install sources to disk
      running 'curtin extract'
        curtin command extract
          acquiring and extracting image from cp:///tmp/tmpIpa3_u/mount \
```

See p 6 for details

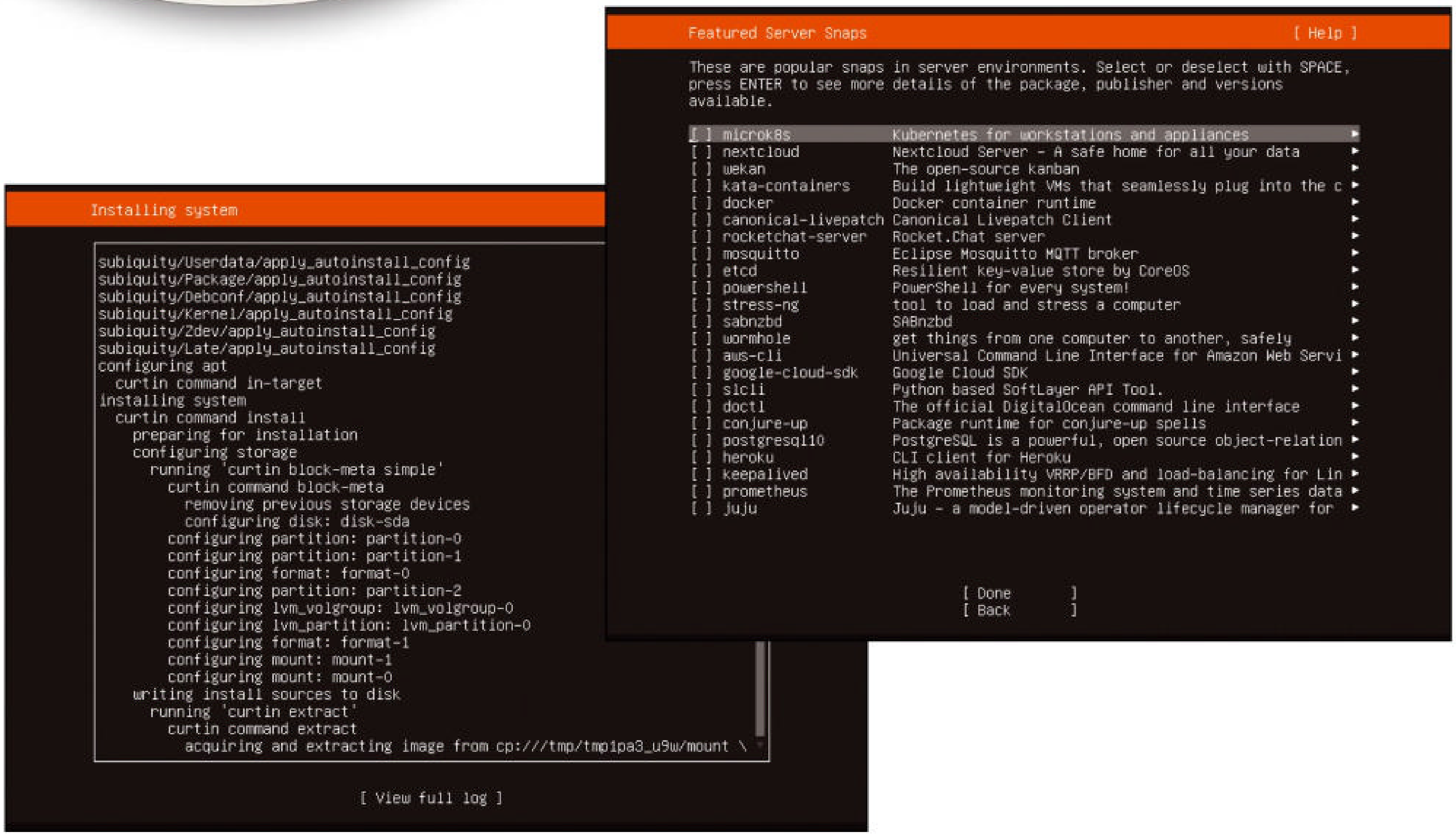

ubuntu 22.04
"Jammy Jellyfish" LTS Server Edition
ISSUE 69/2022
ADMIN
Network & Security
While this ADMIN magazine disc has been tested and is to the best of our knowledge free of malicious software and defects, Admin Magazine cannot be held responsible and is not liable for any disruption, loss, or damage to data and computer systems related to the use of this disc.
DVD ROM

Ubuntu Server 22.04 LTS (Install)

On the DVD

The Ubuntu Server 22.04 (Jammy Jellyfish) long-term support (LTS) release is suitable for enterprise-class deployments "from the data center to the edge" [1] and will be supported until 2032. This latest release offers hardened, compliant, and secure cloud images for Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP), as well as trusted OCI images for Kafka, Grafana, and Loki. Other features include:

- Compatibility with all major architectures
- OpenSSL 3.0
- Linux 5.15 kernel
- SmartNIC support in Netplan



```
Featured Server Snaps                                                [ Help ]

These are popular snaps in server environments. Select or deselect with SPACE,
press ENTER to see more details of the package, publisher and versions
available.

[ ] microk8s            Kubernetes for workstations and appliances           ▸
[ ] nextcloud           Nextcloud Server - A safe home for all your data     ▸
[ ] wekan               The open-source kanban                               ▸
[ ] kata-containers     Build lightweight VMs that seamlessly plug into the c ▸
[ ] docker              Docker container runtime                             ▸
[ ] canonical-livepatch Canonical Livepatch Client                           ▸
[ ] rocketchat-server   Rocket.Chat server                                   ▸
[ ] mosquitto           Eclipse Mosquitto MQTT broker                        ▸
[ ] etcd                Resilient key-value store by CoreOS                  ▸
[ ] powershell          PowerShell for every system!                         ▸
[ ] stress-ng           tool to load and stress a computer                   ▸
[ ] sabnzbd             SABnzbd                                              ▸
[ ] wormhole            get things from one computer to another, safely      ▸
[ ] aws-cli             Universal Command Line Interface for Amazon Web Servi ▸
[ ] google-cloud-sdk    Google Cloud SDK                                     ▸
[ ] slcli               Python based SoftLayer API Tool.                     ▸
[ ] doctl               The official DigitalOcean command line interface     ▸
[ ] conjure-up          Package runtime for conjure-up spells                ▸
[ ] postgresql10        PostgreSQL is a powerful, open source object-relation ▸
[ ] heroku              CLI client for Heroku                                ▸
[ ] keepalived          High availability VRRP/BFD and load-balancing for Lin ▸
[ ] prometheus          The Prometheus monitoring system and time series data ▸
[ ] juju                Juju - a model-driven operator lifecycle manager for  ▸

                          [ Done       ]
                          [ Back       ]
```

```
Installing system

subiquity/Userdata/apply_autoinstall_config
subiquity/Package/apply_autoinstall_config
subiquity/Debconf/apply_autoinstall_config
subiquity/Kernel/apply_autoinstall_config
subiquity/Zdev/apply_autoinstall_config
subiquity/Late/apply_autoinstall_config
configuring apt
  curtin command in-target
installing system
  curtin command install
    preparing for installation
    configuring storage
      running 'curtin block-meta simple'
        curtin command block-meta
          removing previous storage devices
          configuring disk: disk-sda
        configuring partition: partition-0
        configuring partition: partition-1
        configuring format: format-0
        configuring partition: partition-2
        configuring lvm_volgroup: lvm_volgroup-0
        configuring lvm_partition: lvm_partition-0
        configuring format: format-1
        configuring mount: mount-1
        configuring mount: mount-0
    writing install sources to disk
      running 'curtin extract'
        curtin command extract
          acquiring and extracting image from cp:///tmp/tmp1pa3_u9w/mount \

                       [ View full log ]
```

Resources

[1] Ubuntu Server 22.04 LTS: [https://ubuntu.com/engage/ubuntu-server-22-04-LTS]

[2] Documentation: [https://help.ubuntu.com/]

[3] Installation guide: [https://ubuntu.com/tutorials/install-ubuntu-server#1-overview]

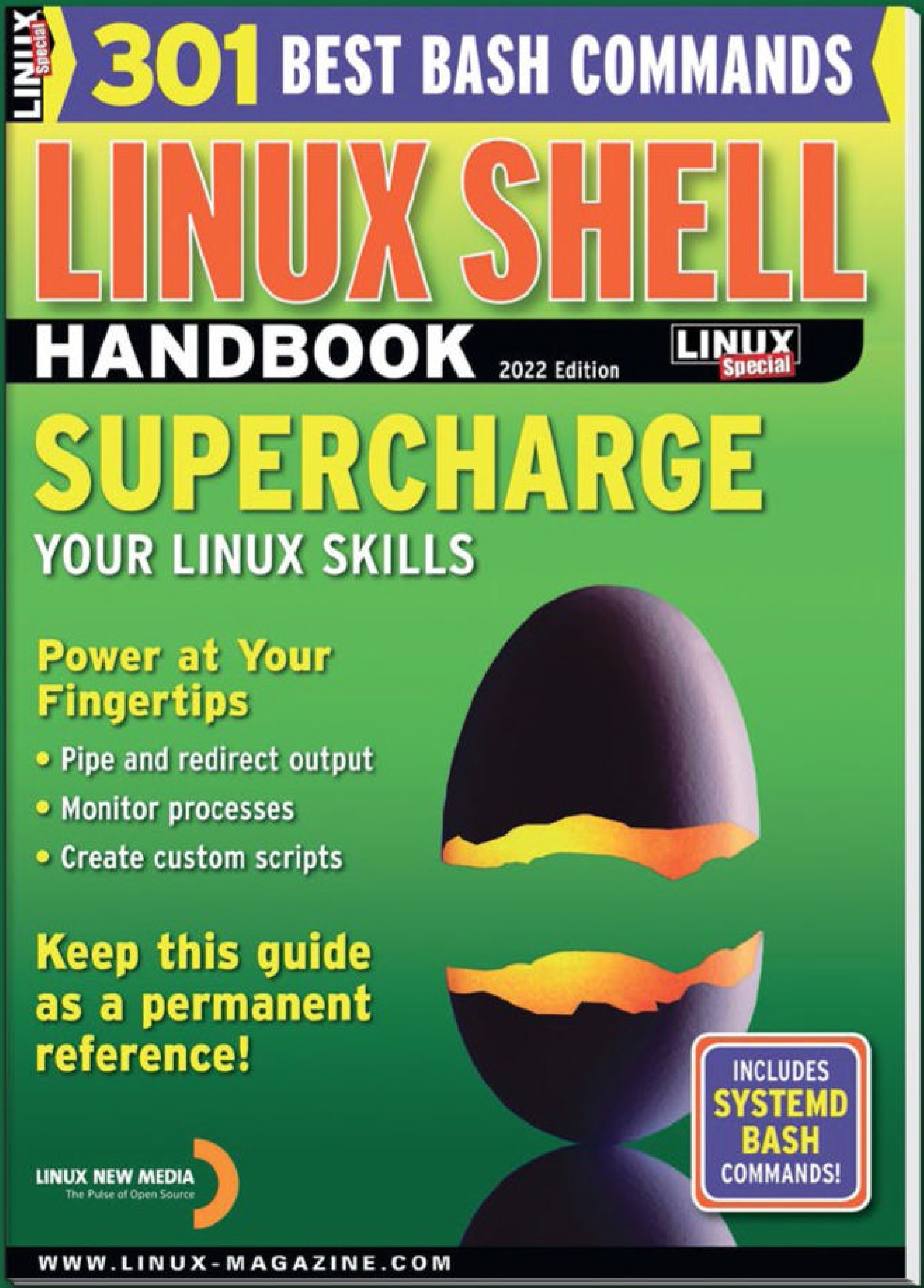
[4] Livepatch Service: [https://ubuntu.com/tutorials/enable-the-livepatch-service#1-overview]

[5] Downloads: [https://ubuntu.com/download/server]

DEFECTIVE DVD?

Defective discs will be replaced, email: cs@admin-magazine.com

While this *ADMIN* magazine disc has been tested and is to the best of our knowledge free of malicious software and defects, *ADMIN* magazine cannot be held responsible and is not liable for any disruption, loss, or damage to data and computer systems related to the use of this disc.


LINUX Special
301 BEST BASH COMMANDS
LINUX SHELL
HANDBOOK 2022 Edition
LINUX Special
SUPERCHARGE
YOUR LINUX SKILLS
Power at Your Fingertips
• Pipe and redirect output
• Monitor processes
• Create custom scripts
Keep this guide as a permanent reference!
INCLUDES SYSTEMD BASH COMMANDS!
LINUX NEW MEDIA
The Pulse of Open Source
WWW.LINUX-MAGAZINE.COM

News for Admins

Tech News

Huge DDoS Attack over HTTPS is Discovered and Stopped

The security company Cloudflare has announced that it detected and mitigated a 15.3 million request-per-second (rps) denial of service attack, which the company called “one of the largest HTTPS attacks on record.” Although larger attacks have occurred on the open Internet, mounting a DDoS attack over HTTPS encrypted connections requires significantly more resources, which means that the scope of this attack is quite remarkable. According to Cloudflare, the botnet used for the attack consisted of 6,000 unique bots from 1,300 different networks in 112 countries.

In recent years, attackers have begun to employ IoT devices in their botnets, leading to a vast increase in the number of available devices. At the same time, the extortion market has become more lucrative. In a typical scenario, attackers will launch a DDoS attack, then force the network owner to pay a ransom to stop the attack. In this case, Cloudflare was able to thwart the attack using a signature-based approach to analyzing the traffic and stopping requests that appeared to be part of the attack.

© Amy Walters, 123RF.com

For more information, see the blog post (https://blog.cloudflare.com/15m-rps-ddos-attack/) on the Cloudflare website.

Canonical Offering a Beta Version of a Real-Time Kernel

The release of Ubuntu 22.04 (Jammy Jellyfish) has been met with almost universal praise. But the company sees certain use cases and scenarios that could greatly benefit from a real-time kernel. This insight has led Canonical to release a beta version built with the out-of-tree PREEMPT_RT patches included. This release is available for x86_64 and AArch64 hardware and is aimed at industries such as communication and manufacturing, that place a high priority on low latency.

Dan Lynch, Marketing Director at Intel, said of the new release, “Ubuntu 22.04 LTS’s real-time kernel unlocks low-latency use cases for real-time applications like Cloud RAN.” Lynch added, “Together with Canonical, we have validated Intel’s FlexRAN SDK to enable OpenRAN implementations requiring pre-emptive real-time kernel capabilities to meet 5G latency requirements.”

With the PREMPT_TR patchset, the real-time kernel enables businesses and developers to leverage Ubuntu for extreme latency-dependent use cases and provides deterministic response times to service events. This new release also makes it possible for companies to rely on the same platform.

Lead Image © vlastas, 123RF.com

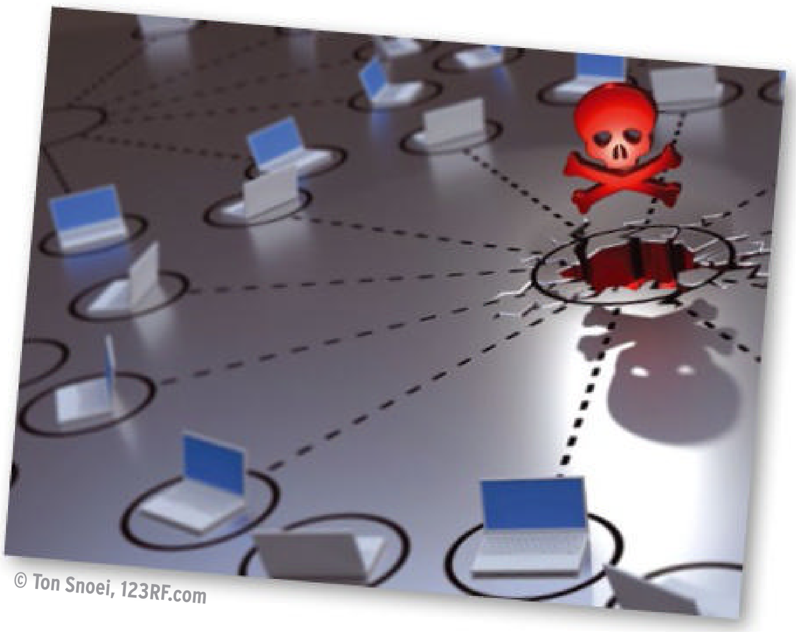
Or, as Radoslaw Adamczyk, Technical Lead at IS-Wireless says, "Now we have one platform for the whole stack, from bare metal with MAAS to Ubuntu OS, LXD VM, and Microk8s on the edge – tested, validated, verified, and secure."

To gain access to the real-time kernel version of Ubuntu 22.04, you must subscribe to Ubuntu Advantage for Infrastructure (https://ubuntu.com/advantage).

A New Linux Vulnerability Could Provide Root Access to Systems

Dubbed "Nimbuspwn," the vulnerabilities (CVE-2022-29799 (https://cve.mitre.org/cgi-bin/cve-name.cgi?name=CVE-2022-29799) and CVE-2022-29800 (https://cve.mitre.org/cgi-bin/cve-name.cgi?name=CVE-2022-29800)) are located in the networkd-dispatcher daemon that checks for systemd-networkd connection status changes.

According to Microsoft's principal security researcher, Jonathan Bar Or (https://www.linkedin.com/in/jonathan-bar-or-89876474/), "Reviewing the code flow for networkd-dispatcher revealed multiple security concerns, including directory traversal, symlink race, and time-of-check-time-of-use race condition issues, which could be leveraged to elevate privileges and deploy malware or carry out other malicious activities."



© Ton Snoei, 123RF.com

Nimbuspwn allows for attackers to deploy payloads (such as a root backdoor) and can be exploited as a vector for root access by attackers using ransomware to reach an even greater impact on vulnerable devices.

The one caveat to Nimbuspwn is that attackers would need local access to targeted systems in order to gain any leverage via the vulnerabilities.

Both vulnerabilities have been patched by the network-dispatcher maintainer, Clayton Craft. All Linux admins are encouraged to immediately update all of their systems to apply the patch.

Mike Parkin, senior technical engineer at Vulcan Cyber said of Nimbuspwn, "Any vulnerability that potentially gives an attacker root-level access is problematic. Fortunately, as is common with many open-source projects, patches for this new vulnerability were quickly released."

Microsoft Blocks VBA Macros from Untrusted Sources

Microsoft has announced (https://docs.microsoft.com/en-us/deployoffice/security/internet-macros-blocked) that it will start blocking Visual Basic for Applications (VBA) macros from untrusted sources by default in future versions of Microsoft Office. VBA macros are often used by attackers as a means for deploying ransomware and other forms of malware. With this change, users will receive a warning when they click on an Office file from an untrusted source that contains macros. The user will then have to make an active choice whether to allow the macros. Many organizations already enable the "Block macros from running in Office files from the Internet" policy, which has a similar effect.

This change only applies to untrusted files. If the file comes from a trusted source, if trust is proven through a digital signature, or if the document was previously marked as trusted, VBA macros are enabled. According to Microsoft, the change will occur in Version 2203 software, which started to roll out in Preview form earlier this month.

This change marks an acknowledgment from Microsoft that untrusted VBA macros are a significant security issue for Windows users. Even if you don't plan to download the new Office version anytime soon, it is a good idea to block untrusted macros as a policy to close off this common form of attack. On the other hand, if your business provides Office documents with embedded macros for download (for example, sales documents or tech sheets), these files might require changes to be fully usable after this change is in effect.

© alphaspirit, 123RF.com

Terraform multicloud orchestrator version 1.0

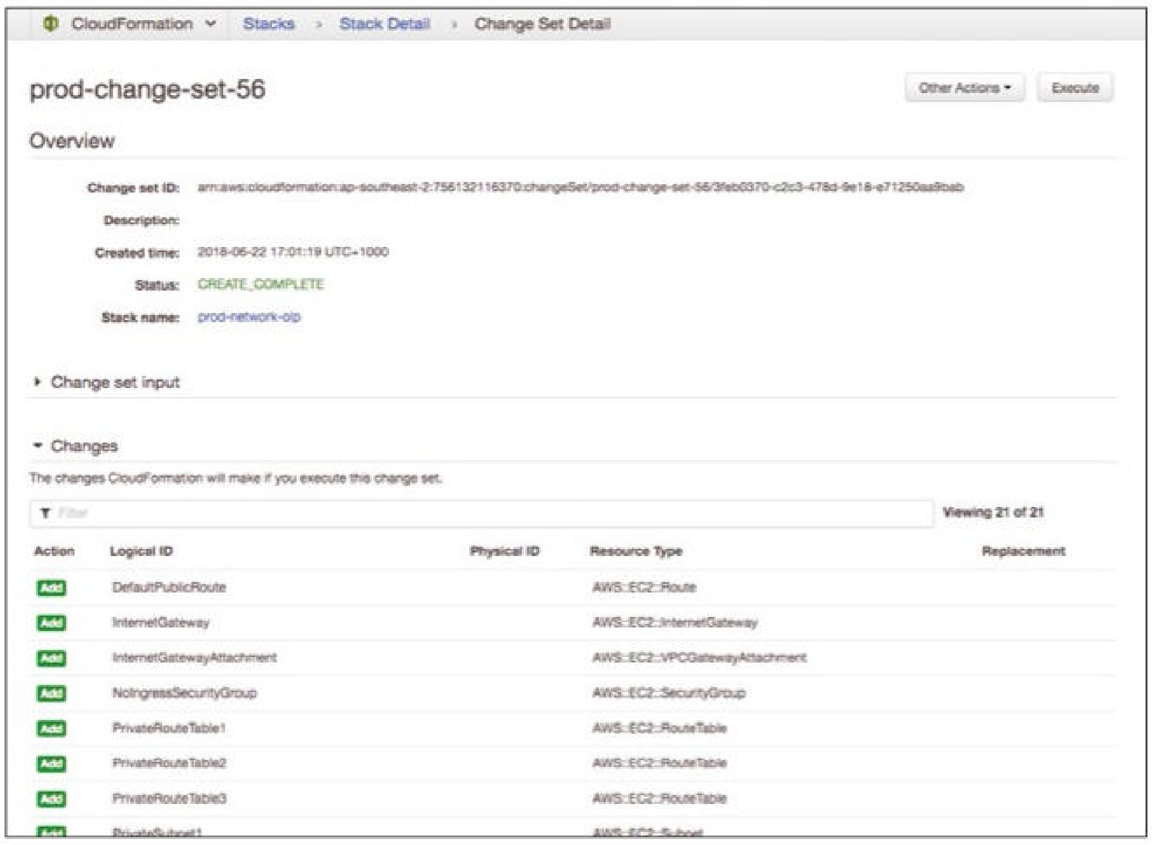
Beautiful Arrangement

HashiCorp's Terraform delivers orchestration for multiple cloud environments and supports a huge set of target platforms. Version 1.0 is considered a milestone. By Martin Loschwitz

Automation is not just "nice to have" in the data center – it is an absolute requirement. The much-cited shortage of skilled employees alone is forcing companies to use automation to free up development resources by letting well-trained employees get on with the interesting work rather than keeping them busy with repetitive tasks. Many admins today don't even bother fully automating their own bare metal, in large part because of the cloud, which seeks to help admins forget all their worries. Yet it is precisely the cloud that impressively shows that many problems don't disappear at all but simply mutate. However, automation in a cloud environment, called orchestration, has to work differently than on bare metal. Orchestration originally arose as a special form of automation, wherein a template file describes how the virtual environment should appear and then feeds this request to a special service for evaluation (Figure 1). The orchestrator then draws on the other cloud services to create the required resources in a completely automatic and mutually coordinated way. Unfortunately each cloud technology has its own templates and syntax that are not exchangeable. Especially in hybrid setups with multiple public cloud approaches, you can very well be forced to do the same work multiple times. This is where Terraform comes in.

Technology Jungle

For years, HashiCorp's tool has touted itself as a powerful and environment-agnostic multicloud orchestrator with its own syntax and an abstraction layer for the major vendors' orchestration APIs. To be sure, Terraform also has its own commands for launching instances in Amazon Web Services (AWS), Azure, or Google Cloud Platform (GCP), so you still do not gain a generic machine resource. However, Terraform offers the huge advantage of only requiring you to learn one set of syntax rather than familiarizing yourself with a wide variety of services. Moreover, the entire infrastructure can be defined as a single continuous directory instead of messing around with custom solutions for Azure, AWS, and the like. A few months ago, the developers released version 1.0 of Terraform,



prod-change-set-56

Overview

Change set ID: arn:aws:cloudformation:ap-southeast-2:756132116370:changeSet/prod-change-set-56/3feb0370-c2c3-478d-9e18-e71250aa9bab

Description:

Created time: 2018-06-22 17:01:19 UTC+1000

Status: CREATE_COMPLETE

Stack name: prod-network-olp

Change set input

Changes

The changes CloudFormation will make if you execute this change set.

Filter — Viewing 21 of 21

Action	Logical ID	Physical ID	Resource Type	Replacement
Add	DefaultPublicRoute		AWS::EC2::Route	
Add	InternetGateway		AWS::EC2::InternetGateway	
Add	InternetGatewayAttachment		AWS::EC2::VPCGatewayAttachment	
Add	NoIngressSecurityGroup		AWS::EC2::SecurityGroup	
Add	PrivateRouteTable1		AWS::EC2::RouteTable	
Add	PrivateRouteTable2		AWS::EC2::RouteTable	
Add	PrivateRouteTable3		AWS::EC2::RouteTable	
Add	PrivateSubnet1		AWS::EC2::Subnet	

Figure 1: Automation in clouds is known as orchestration and is put into practice by services such as AWS CloudFormation. © LevelUp

Photo by Annie Spratt on Unsplash

which they describe as a milestone in the project's history. Now is the perfect opportunity to put Terraform through its paces. How does the tool work, and does it deliver what the vendor promises in terms of cloud orchestration?

Editions

Terraform is open source software at its core, and the basic community version of the software, displayed predominantly on their website [1], forms the basis of the other versions offered by the HashiCorp, including the cloud variant hosted by HashiCorp for cloud users and an Enterprise edition.

The latter version, of course, is aimed at companies that need an on-premises instance of the entire Terraform environment and do not want to see it hosted in a cloud – even though the product the admin gets with Terraform Enterprise is ultimately an on-premises installation of Terraform Cloud. Accordingly, the solutions also share a feature set: Single sign-on (SSO) via Security Assertion Markup Language (SAML) is available in Terraform Cloud and Enterprise, as is the option to store a completely custom layout for networks in the cloud.

Multiclient capability for multiclient software-as-a-service (SaaS) offerings is missing from the classic Terraform, as is extended logging for auditing purposes. The practical GUI for various tasks in Terraform Enterprise is more or less the cherry on top (Figure 2). It is doubtful whether so many pretty accessories are necessary in many environments. Terraform is obviously looking to compete with products such as Ansible Tower or Puppet Enterprise with its cloud and enterprise offerings.

However, the vendor's flowery descriptions on the website cannot hide the fact that Terraform itself (i.e., the core of the solution) has a great deal going for it in the form of open source software and will be all people need in many environments. What can admins expect if they stick with the open source version, and what can you actually do with the product?

Clear-Cut Architecture

When you look at the schematic of the Terraform components, you will notice that the solution is pleasingly simple. If you look at the orchestrators of the major clouds in comparison, you almost always have to deal with various APIs, different daemons, and additional components that all have to be gathered under a single roof.

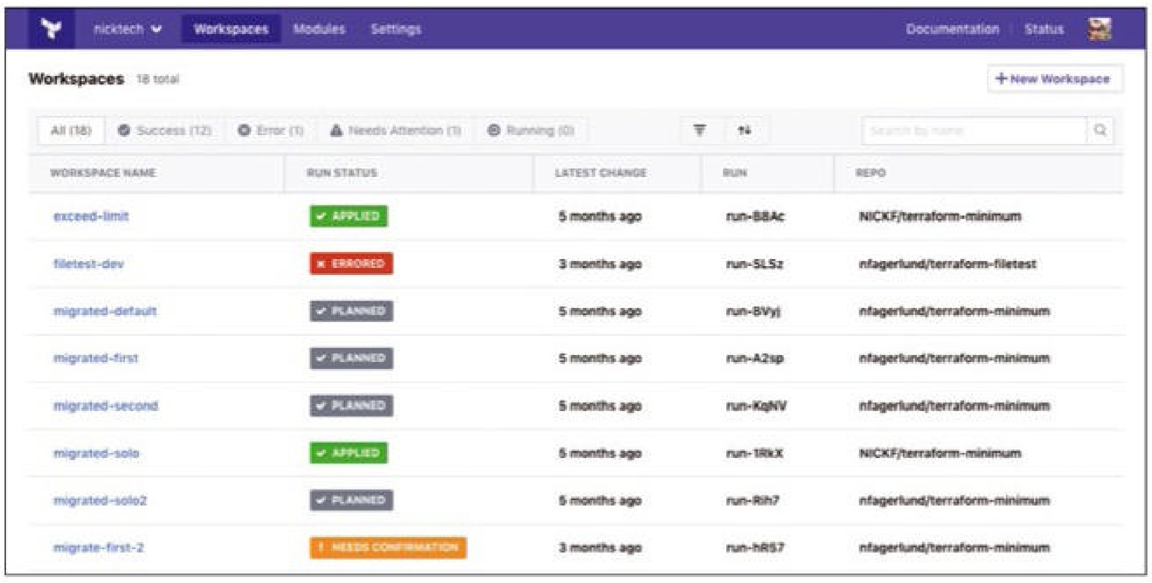
Not so with Terraform and the Terraform Core, which is equivalent to the `terraform` command-line utility. Unlike competing solutions, Terraform does completely without its own APIs or daemons that store arbitrary states. Terraform is not completely stateless, though: Terraform takes care of managing its states itself. Files with a `.tf` file extension let you describe the state that resources in the various clouds need to have. These state files can be stored either locally or in a remote store (e.g., on Amazon's S3). The remote option has the advantage that Terraform can fetch the target state from the network at any time, which means that the program can be called from any host.

Helpful Syntax

To be completely agnostic with regard to the many different target platforms, the Terraform developers have come up with their own declarative configuration language, also known as Terraform for short, which can be used natively as well as in a form converted to JSON.

At its core, the language comprises a small set of basic entry types. A block is a bracket that groups one or more resources and the parameters associated with them. Listing 1 shows a complete example of a block that references another resource (`aws_network_interface`). To turn the example into a complete, working example, you would need to create an additional `aws_network_interface` block, which in turn would need to reference at least the `aws_vpc` as well as `aws_subnet` resources.

The resources in Terraform reference each other in a nested way, but the syntax never becomes so complex that it becomes unmanageable. The example shown here also makes it possible to understand the syntax of a Terraform block. A `resource` is always the type of resource to which the respective entry refers. Terraform



Workspaces 18 total

WORKSPACE NAME	RUN STATUS	LATEST CHANGE	RUN	REPO
exceed-limit	✓ APPLIED	5 months ago	run-B8Ac	NICKF/terraform-minimum
filetest-dev	✗ ERRORED	3 months ago	run-SLSz	nfagerlund/terraform-filetest
migrated-default	✓ PLANNED	5 months ago	run-BVyj	nfagerlund/terraform-minimum
migrated-first	✓ PLANNED	5 months ago	run-A2sp	nfagerlund/terraform-minimum
migrated-second	✓ PLANNED	5 months ago	run-KqNV	nfagerlund/terraform-minimum
migrated-solo	✓ APPLIED	5 months ago	run-1RkX	NICKF/terraform-minimum
migrated-solo2	✓ PLANNED	5 months ago	run-Rih7	nfagerlund/terraform-minimum
migrate-first-2	! NEEDS CONFIRMATION	3 months ago	run-hRS7	nfagerlund/terraform-minimum

Figure 2: The *Workspaces* view, which can be used to centralize management of various projects, is reserved for Terraform cloud variants and the Enterprise on-premises version. © HashiCorp

Listing 1: Terraform Block for AWS

```
resource "aws_instance" "foo" {
  ami          = "ami-005e54dee72cc1d00"
 # us-west-2
 instance_type = "t2.micro"
 network_interface {
 network_interface_id = aws_network_interface.foo.id
 device_index        = 0
 }
 credit_specification {
 cpu_credits = "unlimited"
 }
}
```

uses its provider modules to identify resources (more on that subject later). The resource type to be called is backed up by an identifier, which needs be defined for each resource in Terraform and must be globally unique.

The Terraform language is not so complicated that you can't learn it relatively quickly and is one of HashiCorp's unique selling points, in addition to its versatility. These advantages immediately become apparent when you look at the second central component of Terraform: the modules.

Pooling Resources

What sounds complex is actually relatively simple. The example from Listing 1, for example, covers just a part of the definitions needed to launch a virtual instance on Amazon with a virtual network by Terraform. On the other hand, you need the resources for the network and, if so desired, for persistent storage. This is where modules in Terraform come into play.

Under the hood, modules combine various API calls in the target environments to make your life easier. To stick with the idiom of the previous description: When you call a module, Terraform derives a set of `resource` statements from it in the background, before proceeding to align the resources.

For a good example of this, you again need to enter the AWS world: The *vpc* module creates a virtual network in AWS and lets you create its subnets in simple notation instead of fiddling with `resource` statements. If you call the *ec2-instance* module after the VPC module in your projects, you can then directly reference the subnet created by the VPC. The two resources can also be linked.

Provider at the Core

This functionality begs the question: If modules only bundle resources, how do they know which ones to bundle? What is the origin of that part of Terraform that ultimately generates the code in Amazon Elastic Compute Cloud (EC2), Azure, and the like from the Terraform scripts? Herein come the providers: They mediate between Terraform on the one hand and the target platforms on the other.

Out of the box, Terraform already comes with a huge selection of providers. The AWS provider, which has already been mentioned several times, is certainly the king of the hill. That said, the providers for Azure or Google's GCP offer a similarly high level of functionality. You would be mistaken to assume that Terraform can only be used to manipulate resources in cloud environments, though. In the meantime, Terraform's abstraction capability is so massive that other tasks can also be implemented completely without reference to the cloud. If you use this type of provider, Terraform is more of an automation solution such as Puppet or Chef in some ways, with a complexity that can be compared with that of Ansible.

If you look at the list of providers [2] at Terraform, it quickly becomes clear that the HashiCorp is not holding back. Terraform maintains a public directory of all available providers in the form of the registry, a kind of public marketplace (Figure 3). In addition to the providers mentioned for public clouds, you will find providers for OpenStack, for Oracle's public cloud, and for a variety of Kubernetes distributions. Pods can then also be defined directly in Kubernetes by Terraform, or containers can be managed in a fully automated process across the nodes of a fleet manager.

If you use content delivery networks (CDNs) such as Akamai or Cloudflare, you can also manipulate and control them with Terraform, as you can databases: With the use of different providers, you can execute actions directly in MongoDB or MariaDB, for example. Worth mentioning for fun: You can use a Terraform provider to order pizzas from the US pizza delivery service Domino's.

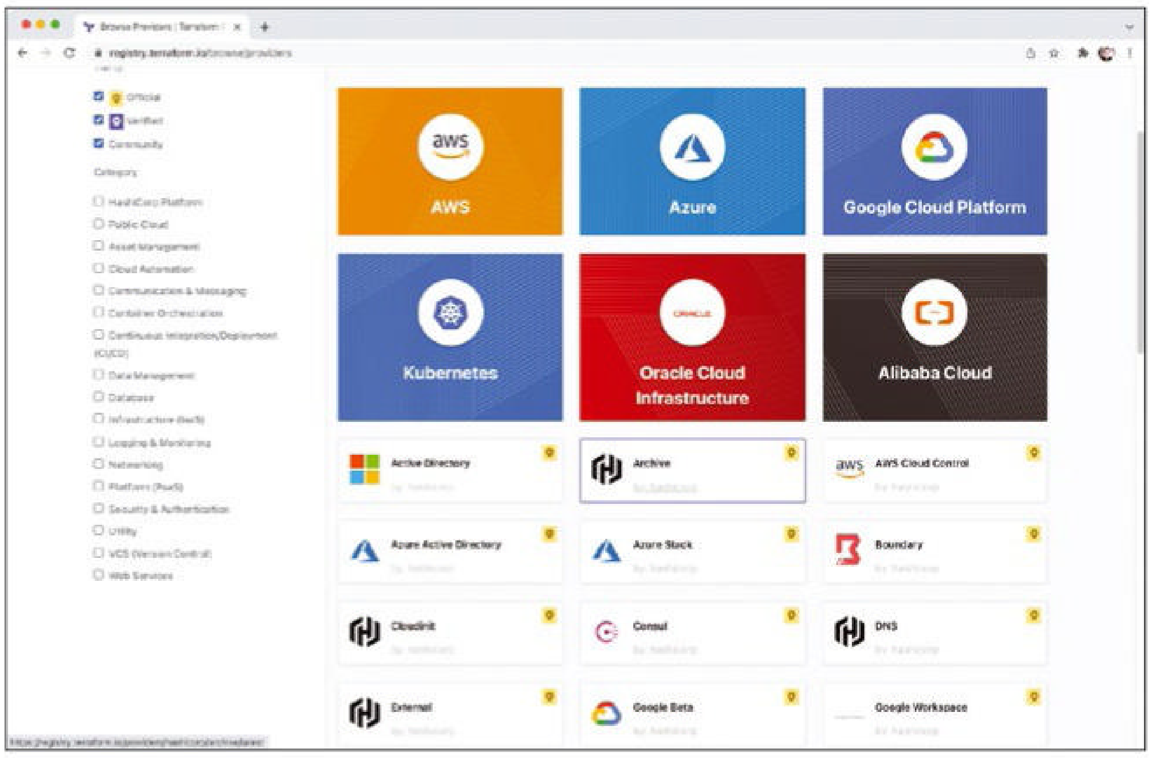


Figure 3: HashiCorp operates a registry for providers that acts as a kind of public marketplace.

Working with Terraform

In addition to the easily understandable configuration language, the simplicity of Terraform's operations is pleasing. Apart from `terraform`, you need nothing else to get started. How exactly does this work look?

One central theme in the Terraform world is the plan, short for the "execution plan," which contains all the steps you want Terraform to carry out. Enabling a plan comprises several steps; fortunately, they are quite simple and occur in a logical sequence.

The first step is to create a `.tf` file, which, as already mentioned, creates the state you want the installation to achieve. The `.tf` file lists the modules and providers to be used. Although you can find many prebuilt modules for Terraform online, you might need a local module for special functions. If need be, you can include your local modules in a state file directly through the local filesystem.

For any files that do not already exist locally, the `terraform init` command brings in all the modules and providers that Terraform needs. After that, the `terraform apply` command sends Terraform Core on its way to roll out the resources described in the state file. The plan is then considered executed and the resources are running.

Terraform 1.0

As mentioned before, the Terraform developers recently released version 1.0 of their software. Anyone familiar with the release cycles of other solutions would expect massive innovations and quantum leaps in the feature set at this point, but that is not the point of Terraform 1.0. In an approach similar that frequently practiced by Linux inventor Linus Torvalds, the Terraform developers simply assign the 1.0 version number to a release of the 0.15 branch at some point.

After more than 10 years of development, HashiCorp finally considers its own protégé ready for production. Moreover, Terraform 1.0 does not require you to migrate from previous versions. Anyone using 0.15 can switch to 1.0 without fear of failure.

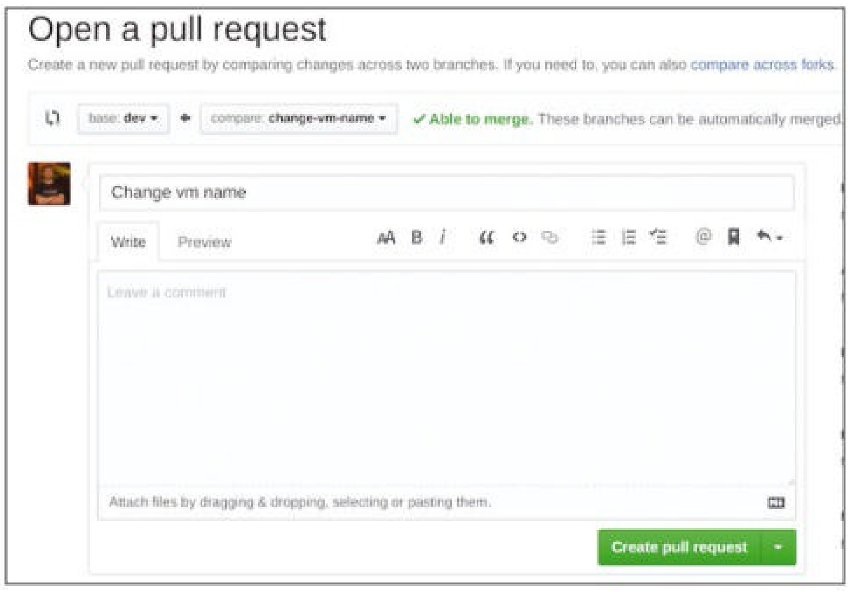
A few key differences are evident between Terraform 0.15 and 1.0, though. The developers promise significantly longer maintenance cycles for Terraform 1.0 than for previous versions. Bug and vulnerability fixes for the 1.0 branch will take place for at least 18 months.

CI/CD Integration

An article about Terraform 1.0 would not be complete without mentioning the solution's advanced continuous integration and continuous delivery (deployment) (CI/CD) integration. This practice is especially important for workloads in clouds and containers. If you already implement infrastructure as code (IaC) (i.e., you describe the environment in the Terraform language), then you will probably also want tests and automatic production deployment when you make a change to the Git directories containing the environment description. Terraform, especially in the form of its cloud products, offers interfaces to GitHub (Figure 4). However, DIY solutions based on Jenkins are not a problem.

Setups like those described make life easier for admins. If you change the description of the infrastructure in the template, GitHub first sends a corresponding message to Terraform. Terraform immediately performs a syntax check and checks whether the new resource specification is plausible and can be implemented in the cloud. After successfully completing all these steps, Terraform then executes an `apply` for the new plan at the end of the pipeline, bringing the environment into line with the target. Of course, Terraform is designed out of the box to make as few changes as necessary to the running resources.

For example, if you change the configuration of a database-as-a-service (DBaaS) instance in AWS, Terraform does not clean it up in the context of a CI/CD pipeline and create it with a new configuration. Instead, Terraform adapts the configuration of the running instance to bring it back into compliance – or outputs an error message if that doesn't work.


Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

base: dev ▾ compare: change-vm-name ▾ ✓ Able to merge. These branches can be automatically merged.

Change vm name

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Figure 4: If you combine Terraform with a pipeline for CI/CD environments such as Jenkins, Git commits immediately trigger changes to running resources on demand. © Google

The State Engine

One internal component of Terraform that has not been sufficiently investigated thus far, and one that system administrators often criminally neglect, is the tool's internal state engine. An IaC solution doesn't just involve matching resources in the cloud to the admin's specifications. Instead, Terraform also has to keep a record of when it made what changes and who instructed it to do so. The command `terraform state show` gives you immediate insights into the individual resource records it has created.

Who Needs Terraform?

At this point, I'll briefly look at who could benefit from Terraform as a solution. The program is by far not the only orchestrator for IaC, but it is certainly one of the most complete.

As usual, great flexibility comes at the price of at least slightly more complexity. If you only want to launch a single EC2 instance on AWS, the single Terraform state file you produce will be very short. In this case, it might not be worthwhile using Terraform at all. Examples of how to achieve the same effect with Amazon's CloudFormation are abundant online. The same applies to classic automators: An EC2 instance can also be started up quickly by Ansible, for example.

More often than not, Terraform will be used in large environments as part of a fairly complex CI/CD pipeline that exploits IaC down to the last detail. Terraform does this better than any other program currently available on the market.

Conclusions

The developers are deliberately not pushing Terraform 1.0 as an epochal release with a big party and a fireworks display of new features. Instead, HashiCorp wanted to do justice to nearly 10 years of work on Terraform, which have resulted in a version of the program that genuinely deserves to be designated "ready for production." From a news point of view, Terraform 1.0 does not provide much fodder. From the point of view of system administrators already bitten by the Terraform bug, however, Terraform 1.0 is a blessing: Wherever Terraform 0.15 has been used thus far, Terraform 1.0 can step in without any worries.

In all likelihood, users can look forward to genuine changes again in the context of Terraform 1.1 [3]. The first release candidate was published just before this issue went to press and focuses on the integrity of the rolled-out services. Unlike its predecessor, Terraform 1.1 checks whether the modules and providers used when creating a plan (`applying`) still match the modules available locally, which prevents those who change modules or provider interfaces from inadvertently throwing working resources out with the bath water.

If you need a highly functional tool for multiple cloud orchestration and IaC, you will definitely want to include Terraform in your considerations, especially because potential alternatives are few and far between: Pulumi (Figure 5) is one [4] and Attune (Figure 6) another [5]. As already mentioned, the common automators handle some parts of Terraform's functionality, as well. ■

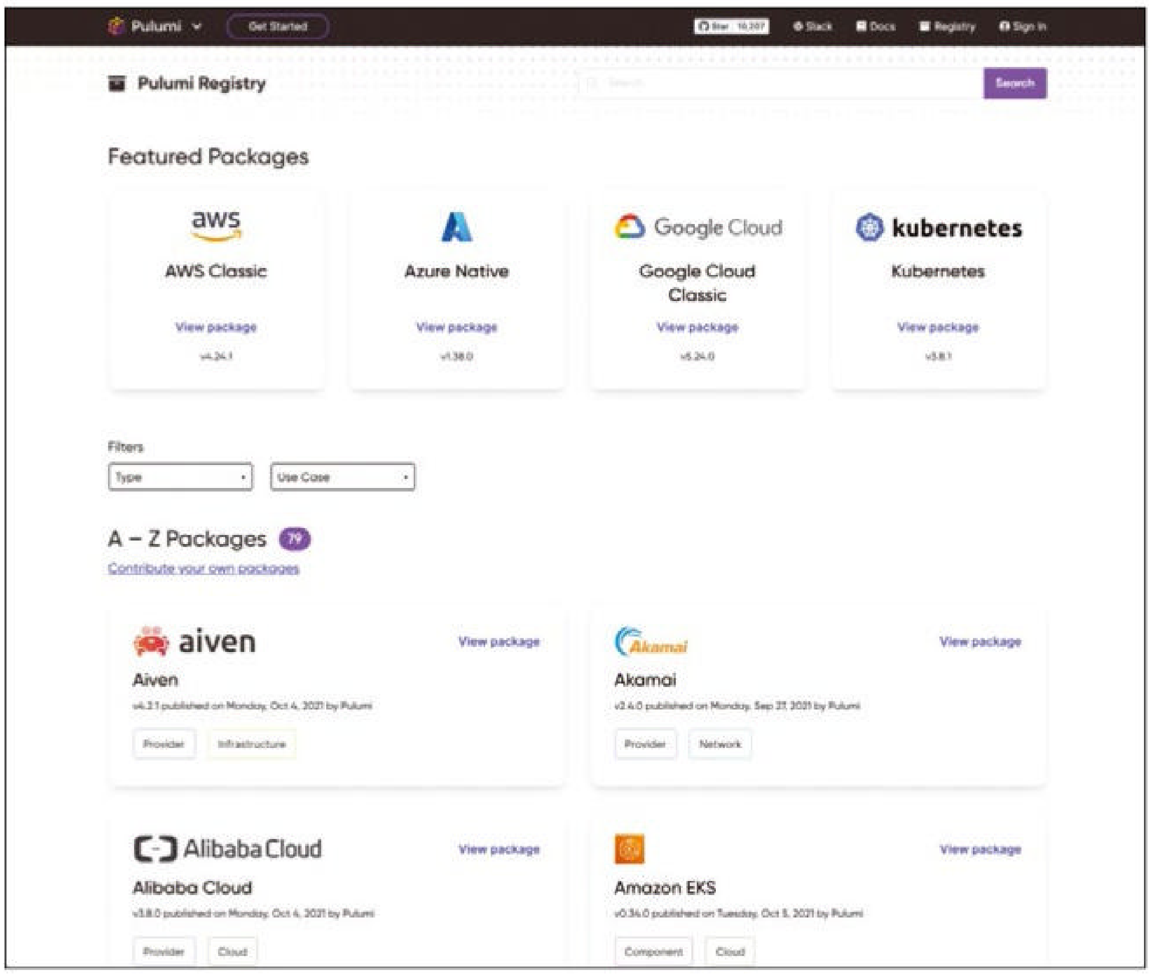


Figure 5: Alternatives to Terraform are few and far between: Only Pulumi, which requires a connection to the cloud ...

Info

[1] Terraform: [https://www.terraform.io]

[2] Terraform providers: [https://registry.terraform.io/browse/providers]

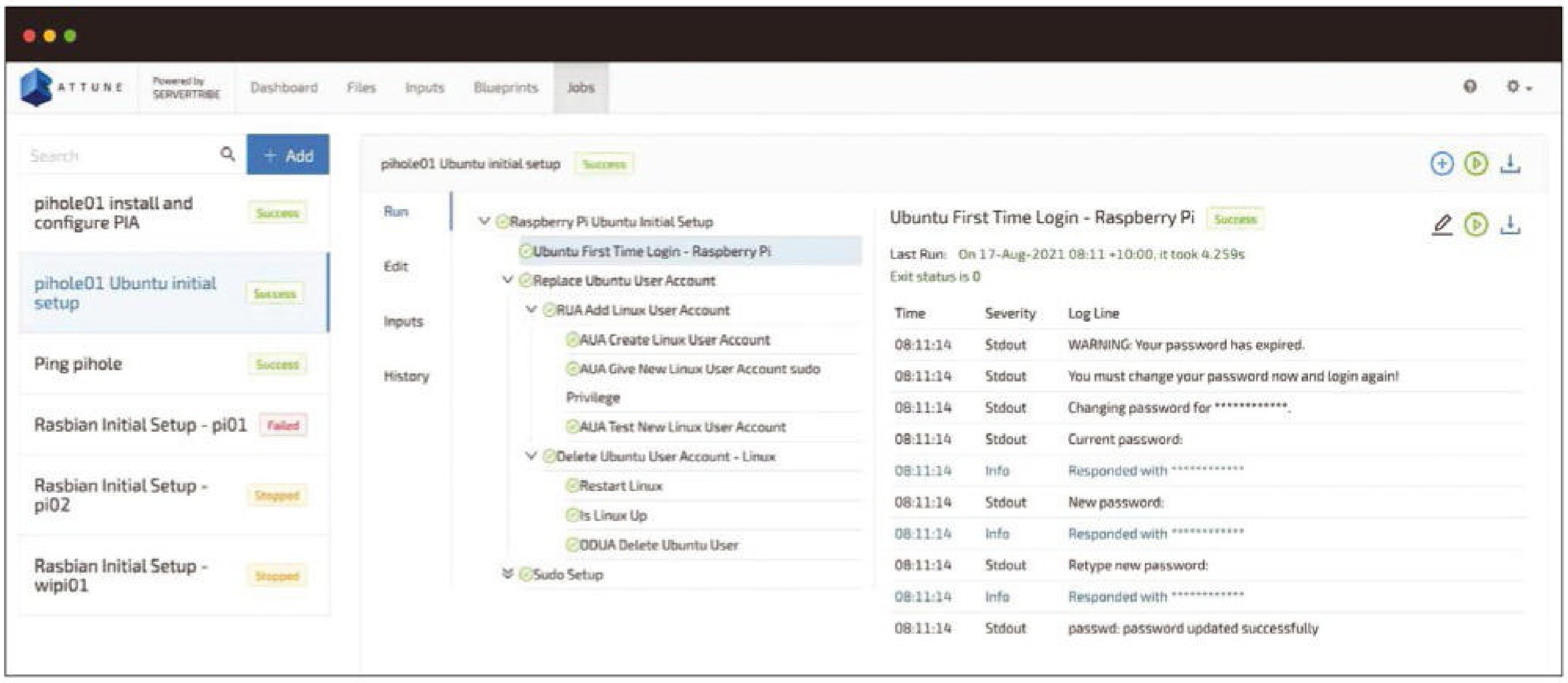
[3] Upgrading to 1.1: [https://www.terraform.io/language/upgrade-guides/1-1]; Terraform latest: [https://www.terraform.io/downloads]

[4] Pulumi: [https://www.pulumi.com]

[5] Attune: [https://www.servertribe.com]

The Author

Freelance journalist Martin Gerhard Loschwitz primarily focuses on topics such as OpenStack, Kubernetes, and Ceph.


ATTUNE
Powered by SERVERTRIBE
Dashboard
Files
Inputs
Blueprints
Jobs
Search
+ Add
pihole01 install and configure PIA
Success
pihole01 Ubuntu initial setup
Success
Ping pihole
Success
Rasbian Initial Setup - pi01
Failed
Rasbian Initial Setup - pi02
Stopped
Rasbian Initial Setup - wipi01
Stopped
pihole01 Ubuntu initial setup
Success
Run
Edit
Inputs
History
Raspberry Pi Ubuntu Initial Setup
Ubuntu First Time Login - Raspberry Pi
Replace Ubuntu User Account
RUA Add Linux User Account
AUA Create Linux User Account
AUA Give New Linux User Account sudo Privilege
AUA Test New Linux User Account
Delete Ubuntu User Account - Linux
Restart Linux
Is Linux Up
DDUA Delete Ubuntu User
Sudo Setup
Ubuntu First Time Login - Raspberry Pi
Success

Last Run: On 17-Aug-2021 08:11 +10:00, it took 4.259s
Exit status is 0

Time	Severity	Log Line
08:11:14	Stdout	WARNING: Your password has expired.
08:11:14	Stdout	You must change your password now and login again!
08:11:14	Stdout	Changing password for ************.
08:11:14	Stdout	Current password:
08:11:14	Info	Responded with ************
08:11:14	Stdout	New password:
08:11:14	Info	Responded with ************
08:11:14	Stdout	Retype new password:
08:11:14	Info	Responded with ************
08:11:14	Stdout	passwd: password updated successfully

Figure 6: ... and Attune are available, with Attune having a different focus and offering significantly less in terms of functionality. © Servertribe

CIO/CISO DACH SUMMIT

#CIOCISODACHSummit | June 21st, 2022 | In-Person, Frankfurt

Create a Game-Changing IT and Security Strategy to Accelerate Your Business

The global pandemic has changed the way we all live and work and now more than ever the IT/security department has become crucial in keeping businesses growing and employees connected. One of the unexpected, positive, changes for CIOs, CISOs and senior tech executives has been the rapid shift in digital technology to ensure businesses can stay ahead of their competitors and discover new opportunities to drive business growth whilst lowering costs, as we attempt to get back to normality.

You've led your team through one of the hardest periods in our global history and overcome huge obstacles for not only your IT and security department but your entire business, but all of these challenges also bring many possibilities. This is why attending the CIO/CISO DACH Summit will give you the chance to engage in peer-to-peer networking, whilst discussing the issues currently affecting senior executives from a variety of key industries.

REGISTER NOW

SNEAK PEEK OF SPEAKERS:

Chuks Ojeme
CISO
Brenntag

Dr. Anke Sax
COO & CTO
KGAL GmbH & Co.

Gerald Scheurmann-Kettner
CIO
EVENT Hotels

Dr. Katja Nettesheim
CEO
Culcha

Florian Jörgens
CISO Group
Corporate Governance
Vorwerk

KEY TOPICS INCLUDE:

- *Creating A Culture Of Holistic Cybersecurity*
- *Transformation Of The Organisation To A Product Centric Structure & Focus*
- *Why Zero Trust Is The World's Only True Cybersecurity Strategy*
- *The CIO vs. The CISO*
- *People, Process, & Platforms; Tech Must Change Before It's Too Late!*
- *How To Lead Your Organisation Through Digital Transformation*
- *How Hackers Can Help You Protect Your Company*
- *Addressing The Technology Skills Gap: Reskilling & Upskilling*
- *Artificial Intelligence In Cybersecurity: Dream vs. Reality*
- *Using Automation To Cure Vulnerability Management*
- *Balancing an Ecosystem of Risk, Security and Trust*
- *How to Create Value in a Fair Data Ecosystem*
- *The Evolving Role Of A CIO & CISO*

For details on how you can attend or to sponsor this event, email:
marketing@cdmmedia.com

CDM
MEDIA

Kubernetes for small and medium-sized enterprises

Advantage Small

We look at the benefits of Kubernetes outside of large corporate environments. By Martin Loschwitz

Hype is not unusual in IT and is initially met with an innate sense of distrust by admins, who tend not to want to deal with all of that new-fangled stuff. This was the case with cloud computing – which many thought was a flash in the pan – as well as during the rise to fame of the fleet orchestrator Kubernetes, which many observers categorically ruled out.

In 2022, it's clear that Kubernetes and containers are here to stay. In many places, the benefits of the solution are so huge that admins quickly turned from container skeptics to container enthusiasts, especially in large companies and corporate environments. According to a survey by SUSE [1], more than 60 percent of companies whose IT budgets exceed EUR10 million per year have cloud-native applications, whereas cloud-native is a distant also-ran among smaller enterprises.

It's easy to get the impression that containers offer little or no added value for smaller companies. But is that really true? Do small and medium-sized enterprises (SMEs) really have nothing to gain from Kubernetes if their IT fleet is smaller (i.e., if they don't have a server fleet of thousands of devices)? In this article, I explore this question and show how even smaller companies can benefit from Kubernetes.

Defining the Terms

To identify the benefits of modern technologies, even for smaller enterprises, it is useful to define the terms clearly. Although terms such as container, Kubernetes, cloud-native, and microarchitecture are mixed up in public discourse and sometimes used as synonyms, one thing is clear: Containers and Kubernetes are not the same thing, and the use of containers does not mandate the use of modern applications that follow the microarchitecture approach.

Therefore, it makes sense to look at containers and fleet management separately to identify the sweet spots for SMEs. Initially, I look exclusively at containers. By definition, a container is simply the filesystem of a (minimal) Linux system in combination with an application installed in it. Depending on the container technology used, you have the option of connecting volumes to a container as persistent storage, which ensures that conventional applications such as databases can be operated in a meaningful way in containers.

Classic Setup

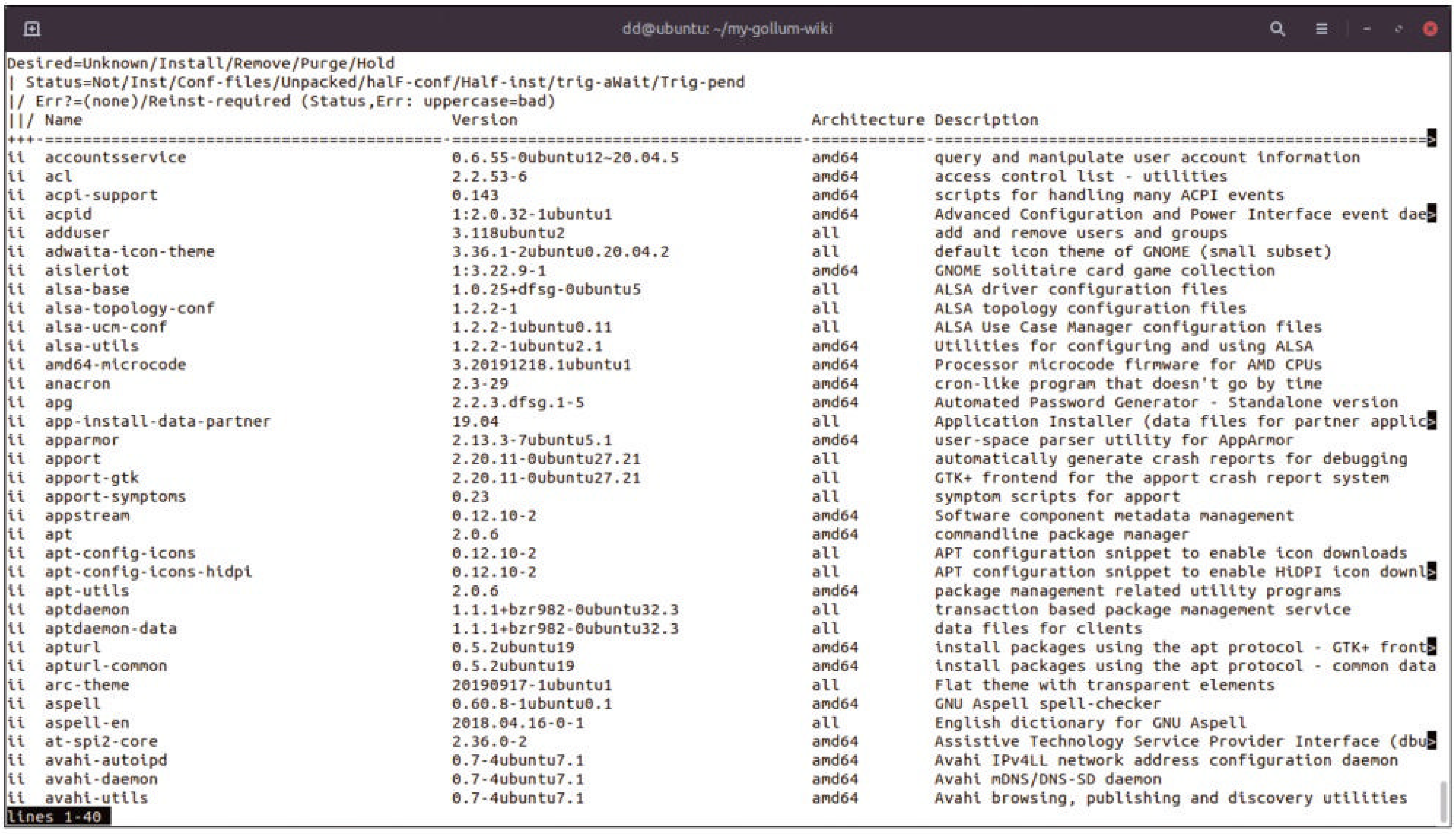
To discuss the practical benefits of containers in everyday life, it helps to take a look at the classic IT setup of the noughties, which is still in use in many places today. Rest assured, it has very little in common with the modern container world.

In these typical setups, each system is assigned a fixed task. Because not even virtualization plays a role in many cases, a static link exists between a system and the application that runs on it. The underpinnings for such a system are almost always one of the classic Linux distributions: Think AlmaLinux or Ubuntu.

Although many companies claim to have achieved a high degree of automation, these kinds of systems are often enough tedious, manually packaged individual installations. Because every modern distribution comes with a package manager (e.g., RPM or DPKG, Figure 1), companies still make extensive use of package managers in many places. The required userland software is therefore installed on the system as a package, and the admin configures it either manually or uses some kind of automation. So far, so familiar.

Every admin with some professional experience has heard the term "dependency hell," which refers to a problem that often occurs on Linux

Lead Image © zelfit, 123RF.com



```
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name                                       Version                               Architecture Description
+++-==========================================-=====================================-============-===================================================>
ii  accountsservice                            0.6.55-0ubuntu12~20.04.5              amd64        query and manipulate user account information
ii  acl                                        2.2.53-6                              amd64        access control list - utilities
ii  acpi-support                               0.143                                 amd64        scripts for handling many ACPI events
ii  acpid                                      1:2.0.32-1ubuntu1                     amd64        Advanced Configuration and Power Interface event dae>
ii  adduser                                    3.118ubuntu2                          all          add and remove users and groups
ii  adwaita-icon-theme                         3.36.1-2ubuntu0.20.04.2               all          default icon theme of GNOME (small subset)
ii  aisleriot                                  1:3.22.9-1                            amd64        GNOME solitaire card game collection
ii  alsa-base                                  1.0.25+dfsg-0ubuntu5                  all          ALSA driver configuration files
ii  alsa-topology-conf                         1.2.2-1                               all          ALSA topology configuration files
ii  alsa-ucm-conf                              1.2.2-1ubuntu0.11                     all          ALSA Use Case Manager configuration files
ii  alsa-utils                                 1.2.2-1ubuntu2.1                      amd64        Utilities for configuring and using ALSA
ii  amd64-microcode                            3.20191218.1ubuntu1                   amd64        Processor microcode firmware for AMD CPUs
ii  anacron                                    2.3-29                                amd64        cron-like program that doesn't go by time
ii  apg                                        2.2.3.dfsg.1-5                        amd64        Automated Password Generator - Standalone version
ii  app-install-data-partner                   19.04                                 all          Application Installer (data files for partner applic>
ii  apparmor                                   2.13.3-7ubuntu5.1                     amd64        user-space parser utility for AppArmor
ii  apport                                     2.20.11-0ubuntu27.21                  all          automatically generate crash reports for debugging
ii  apport-gtk                                 2.20.11-0ubuntu27.21                  all          GTK+ frontend for the apport crash report system
ii  apport-symptoms                            0.23                                  all          symptom scripts for apport
ii  appstream                                  0.12.10-2                             amd64        Software component metadata management
ii  apt                                        2.0.6                                 amd64        commandline package manager
ii  apt-config-icons                           0.12.10-2                             all          APT configuration snippet to enable icon downloads
ii  apt-config-icons-hidpi                     0.12.10-2                             all          APT configuration snippet to enable HiDPI icon downl>
ii  apt-utils                                  2.0.6                                 amd64        package management related utility programs
ii  aptdaemon                                  1.1.1+bzr982-0ubuntu32.3              all          transaction based package management service
ii  aptdaemon-data                             1.1.1+bzr982-0ubuntu32.3              all          data files for clients
ii  apturl                                     0.5.2ubuntu19                         amd64        install packages using the apt protocol - GTK+ front>
ii  apturl-common                              0.5.2ubuntu19                         amd64        install packages using the apt protocol - common data
ii  arc-theme                                  20190917-1ubuntu1                     all          Flat theme with transparent elements
ii  aspell                                     0.60.8-1ubuntu0.1                     amd64        GNU Aspell spell-checker
ii  aspell-en                                  2018.04.16-0-1                        all          English dictionary for GNU Aspell
ii  at-spi2-core                               2.36.0-2                              amd64        Assistive Technology Service Provider Interface (dbu>
ii  avahi-autoipd                              0.7-4ubuntu7.1                        amd64        Avahi IPv4LL network address configuration daemon
ii  avahi-daemon                               0.7-4ubuntu7.1                        amd64        Avahi mDNS/DNS-SD daemon
ii  avahi-utils                                0.7-4ubuntu7.1                        amd64        Avahi browsing, publishing and discovery utilities
lines 1-40
```

Figure 1: **In classic environments, managing software is the package manager's job, which often causes massive dependency issues because packages from different sources collide.**

systems when the admin cannot or does not want to make do with the package pool offered by their choice of distribution and includes external software directories. In many cases this situation is unavoidable.

If you want to have up-to-date software on older distributions, your only approach is typically through the software vendor's own repository, and if the vendor then decides that the collection of packages included in, say, CentOS 7 is not enough and that the extra packages for Enterprise Linux (EPEL) directory are also required, it's the administrator who has to handle it. With an update to a newer version of the Linux distribution, it is not uncommon for systems to blow up in the administrator's face. The parties involved – CentOS, EPEL, and the software vendor – then usually sit back and refute any responsibility.

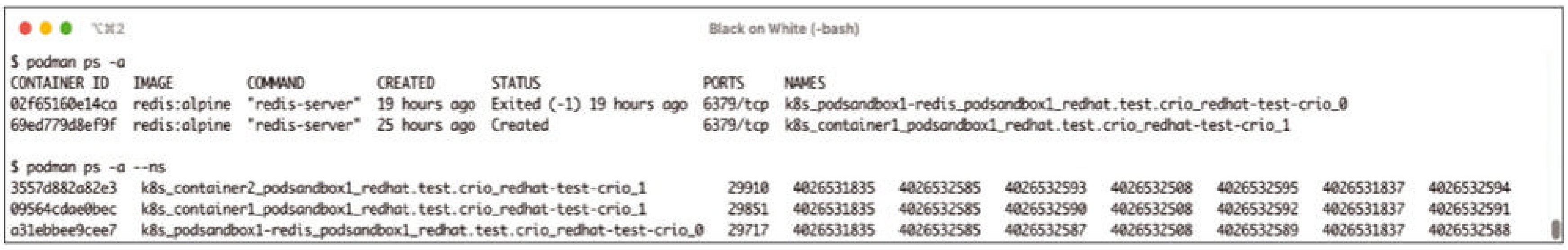
Of course, this is not much use to you as an administrator if you are left without working computers.

Linux distributors recognized this problem several years ago and actively countered it with containers. The big advantage of a container is that it comes with its entire userland in tow. A ready-made container image can be run without any worries on any system that has a runtime for containers – even if it has no additional software, except for the basic components and the container environment.

Because this approach also offers huge advantages for Linux distributors and program providers, the container-based approach has established itself as the de facto standard. Containers offer software providers in particular the ability to deliver their solutions to customers exactly as tested in their own laboratories. They just build one container with a runtime environment for all Linux distributions, instead of a package for every version of every distribution, split into RPM and DEB packages (Figure 2).

Simplification and Greater Efficiency

Simplified management of individual systems should already be incentive enough for small businesses to take a closer look at container technology. After all, if the IT department is not particularly large, you are absolutely dependent on a high level of efficiency. The less overhead the production systems cause in daily life, the better it is from the company's point of view. Containers initially fulfill this



```
$ podman ps -a
CONTAINER ID   IMAGE          COMMAND          CREATED        STATUS                     PORTS      NAMES
02f65160e14ca  redis:alpine   "redis-server"   19 hours ago   Exited (-1) 19 hours ago   6379/tcp   k8s_podsandbox1-redis_podsandbox1_redhat.test.crio_redhat-test-crio_0
69ed779d8ef9f  redis:alpine   "redis-server"   25 hours ago   Created                    6379/tcp   k8s_container1_podsandbox1_redhat.test.crio_redhat-test-crio_1

$ podman ps -a --ns
3557d882a82e3   k8s_container2_podsandbox1_redhat.test.crio_redhat-test-crio_1          29910   4026531835   4026532585   4026532593   4026532508   4026532595   4026531837   4026532594
09564cdae0bec   k8s_container1_podsandbox1_redhat.test.crio_redhat-test-crio_1          29851   4026531835   4026532585   4026532590   4026532508   4026532592   4026531837   4026532591
a31ebbee9cee7   k8s_podsandbox1-redis_podsandbox1_redhat.test.crio_redhat-test-crio_0   29717   4026531835   4026532585   4026532587   4026532508   4026532589   4026531837   4026532588
```

Figure 2: **In container-based environments, applications and their entire userland run in the container. Dependencies no longer play a role, which is not always the case with the package manager for your choice of distribution.**

requirement at the application level. If an application can be rolled out in the shortest possible time without external dependencies so that it only requires a configuration file and persistent memory, this situation translates to significantly lower overhead than installing a package.

Moreover, this procedure facilitates the update process. Administrators periodically update their container workloads by stopping the old container after they have downloaded a new one and connecting the existing persistent volume to a new container launched on the fresh image. Most applications that manage data identify records from a previous version when they encounter them and automatically perform an update, as is the case with MariaDB, for example. Immediately afterward, the new version of the database is up and running. Not only that, if something goes wrong during the update, it is relatively easy to return to the state before the update with the old image and a snapshot of the persistent storage.

Containers also offer simplification and improved efficiency at the system level. If the company's own systems only run containers, many everyday maintenance tasks are eliminated. For example, it no longer has to be a complete CentOS; instead, you can use CoreOS (i.e., a minimal operating system). Although the operating system can't do much apart from running containers, that is exactly what the doctor ordered.

Automation Is Key

For SMEs, moving workloads to containers can even be a stepping stone toward automation. As I mentioned earlier, many admins overestimate the level of automation in their own environments for a number of reasons. Sometimes companies assume (usually wrongly) that automation is not worthwhile because many tasks occur only once. Sometimes, companies are so busy with manual work that they no longer have the resources to look at their automation options, and proudly announced automation projects come to nothing because it turns out that automating legacy systems is far more complicated in practice than it seems to be in theory.

Operating system containers offer an escape route. If your provider can give you a preconfigured container image, you can save yourself much of the application management work your own system involves. Tools such as Ansible make it relatively easy to store a configuration file in the right place in the operating system and then start the container with the correct parameters. Therefore, containers also make implementing automation far easier.

Pared-down operating systems such as CoreOS also have the advantage that automating them is a one-off event that can be repeated as often as you need afterward. Once you have moved the bulk of your applications from bare metal to containers, you will no longer see much of a challenge in creating an AutoYaST or Kickstart configuration for your local environment. As a neat side effect, this move to containers also lets you establish bare metal lifecycle management for the remainder, eliminating manual work for all time and ensuring greater efficiency. This entire workflow operates largely without the constraint of continuous integration and continuous delivery (deployment) (CI/CD) systems, which hover over many a container setup like the Sword of Damocles – and usually without any good reason.

Interim Conclusions

Moving applications into containers clearly delivers added value, especially with a view to maintainability and automation and without the principle of cloud-native applications or Kubernetes even having to be part of the equation. If you regularly spend long hours performing the same manual tasks in your organization, containers in combination with an automator such as Ansible are a good start on the road to improved maintainability and significantly more automation. Lean Linux distributions further reduce the maintenance overhead and offer the opportunity to establish comprehensive bare-metal lifecycle management.

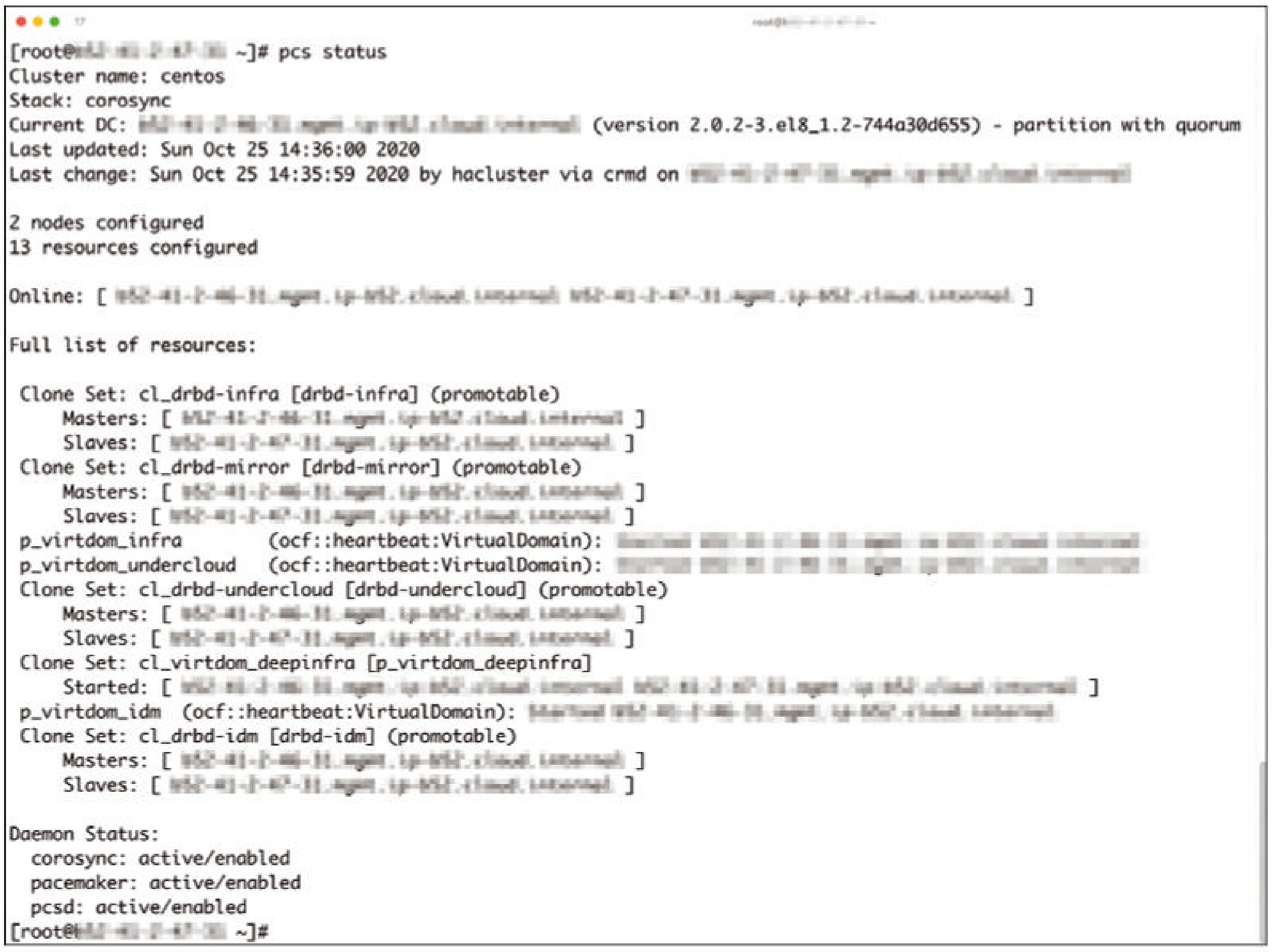
Naturally, this effect is far less pronounced if the application in question is not available as a preconfigured container from your provider. Even if this is the case, though, the work involved with bundling the application into a container and running it yourself still pays dividends in most cases. If you use Docker Hub, make sure that you verify the contents of the containers you use.

What About Kubernetes?

The culmination of the narrative created up to this point is that Kubernetes also has a reason to exist in smaller environments. Assuming you have invested hours of work in packaging a conventional setup in containers, you will soon realize that running containers, much like running classic applications, involves a few challenges. For example, anyone running a database needs high availability (HA). Practically nothing has changed in this respect in the past 20 years. What has changed, however, are the tools that are available to the admin to help implement this kind of setup, which can be easily demonstrated with a simple comparison.

Until the advent of cloud computing and scalable environments, database setups regularly comprised multiple components. The underpinnings were servers, between which the admin created a connection with a cluster manager – usually Pacemaker (Figure 3). A shared storage solution such as a distributed replicated block device (DRBD) took care of making the data available on one node or the other. Together with a virtual IP address, the database then migrated as a resource in Pacemaker from node A to node B, or vice versa, depending on which node had just died.

Setups of this type, however, are not much fun. Even though Pacemaker has become more user friendly in recent years, the software is still highly complex. Most admins will want to steer clear of this kind of setup if they find a way to do so.



```
[root@[redacted] ~]# pcs status
Cluster name: centos
Stack: corosync
Current DC: [redacted] (version 2.0.2-3.el8_1.2-744a30d655) - partition with quorum
Last updated: Sun Oct 25 14:36:00 2020
Last change: Sun Oct 25 14:35:59 2020 by hacluster via crmd on [redacted]

2 nodes configured
13 resources configured

Online: [ [redacted] ]

Full list of resources:

 Clone Set: cl_drbd-infra [drbd-infra] (promotable)
     Masters: [ [redacted] ]
     Slaves: [ [redacted] ]
 Clone Set: cl_drbd-mirror [drbd-mirror] (promotable)
     Masters: [ [redacted] ]
     Slaves: [ [redacted] ]
 p_virtdom_infra        (ocf::heartbeat:VirtualDomain): [redacted]
 p_virtdom_undercloud   (ocf::heartbeat:VirtualDomain): [redacted]
 Clone Set: cl_drbd-undercloud [drbd-undercloud] (promotable)
     Masters: [ [redacted] ]
     Slaves: [ [redacted] ]
 Clone Set: cl_virtdom_deepinfra [p_virtdom_deepinfra]
     Started: [ [redacted] ]
 p_virtdom_idm  (ocf::heartbeat:VirtualDomain): [redacted]
 Clone Set: cl_drbd-idm [drbd-idm] (promotable)
     Masters: [ [redacted] ]
     Slaves: [ [redacted] ]

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
[root@[redacted] ~]#
```

Figure 3: A Pacemaker setup from the good old days but for running virtual machines rather than a database, for example. Components like Pacemaker can easily be replaced by Kubernetes when it comes to running specific services.

Kubernetes as an Alternative

Some system administrators might now claim that Kubernetes can hardly be the answer to Pacemaker's excessive complexity – and there is much truth in this; after all, Kubernetes has its own share of complexity. Although understandable, this argument falls short of the mark. Kubernetes undoubtedly offers a huge amount of functionality, but much of it is irrelevant for the vast majority of setups. As in most cases where a piece of software starts life as something small and manageable but then expands to cover more and more use cases, the Pareto principle applies to Kubernetes: 80 percent of users would be very satisfied with 20 percent of Kubernetes' feature set (i.e., just the basic Kubernetes features).

That Kubernetes can be a genuine alternative, especially if you need a database, is demonstrated by the example described above from the Kubernetes point of view. Two servers are not enough; you need four, but not all of them have to be high-powered beasts with masses of CPU and RAM. Centralized components such as those that belong to Kubernetes can easily be bundled on virtual machines as an alternative. These virtual machines can run in a virtualization cluster that you might even already have in place and applies to the systems that will later run the containers – but you will want sufficient capacity in the form of vCPUs and vRAM.

Which Kubernetes?

The next step is the question as to which Kubernetes distribution to choose for your setup. Red Hat's OpenShift (Figure 4), SUSE's Rancher, and several other variants vie for your favor, but some caution is advised: A single admin with a manageable setup will rarely need OpenShift's feature set. On the downside, a full-blown OpenShift generates a mess of maintenance work, which is probably not worth the while for an SME in most cases.

Instead, it is worth scouring the market for alternatives to major leauge solutions. One good candidate, especially for smaller environments, is MicroK8s [2] from Canonical, void of bells and whistles. Another option might be K3s [3], which claims to be an Internet of Things and edge variant of Kubernetes but still provides a full Kubernetes API and all the relevant functions. Even the upstream version of Kubernetes comes with a corresponding distribution named Minikube [4], which quickly sets up a simple Kubernetes cluster locally.

The decision as to which of these lean systems suits your use case is something you ultimately need to make yourself; again, your own preferences will play a role.

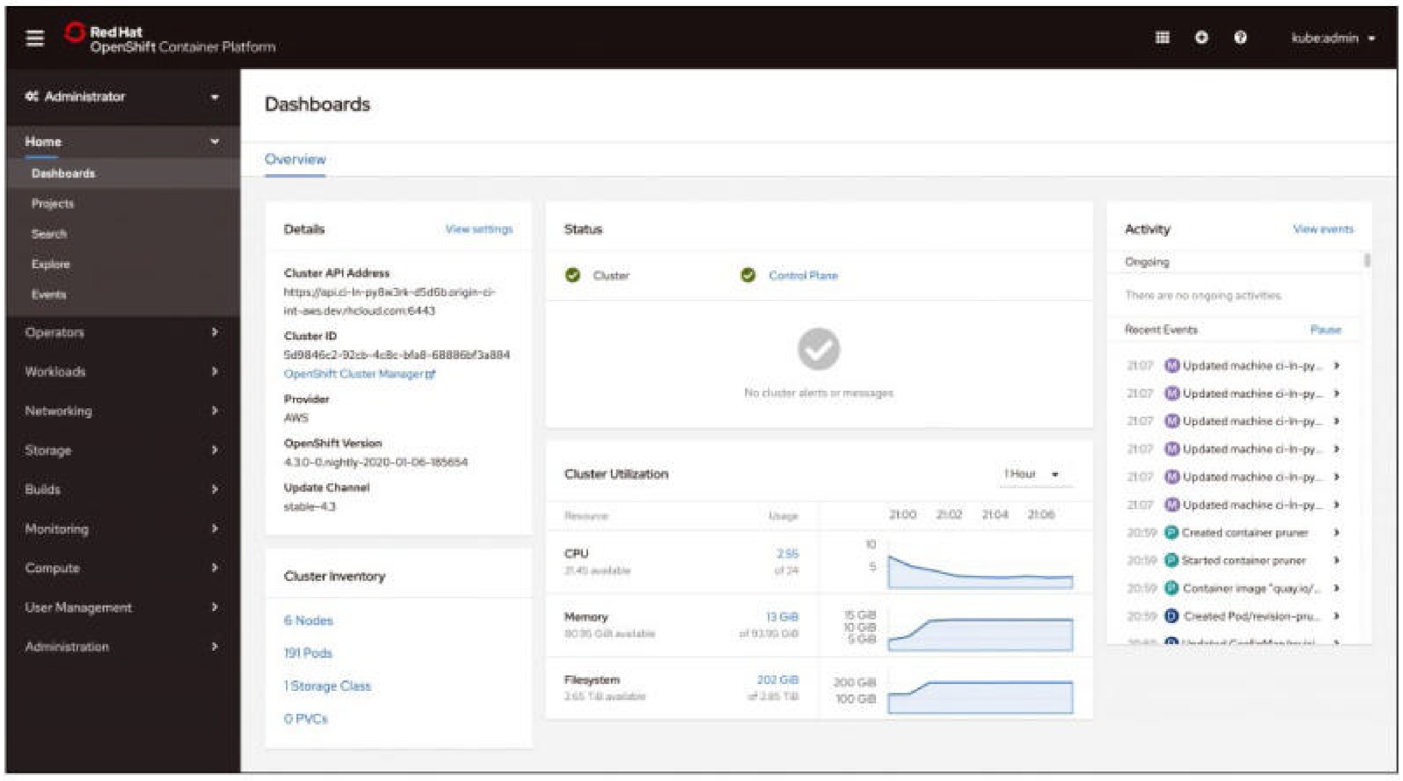
HA MySQL in a Jiff

In a database example, once a Kubernetes cluster is up and running, the remaining steps en route to a highly available MariaDB or MySQL database are quickly taken. The Kubernetes API offers a full set of functions, such as replica sets or stateful sets.

In a setup with Galera (Figure 5), all that is missing is a suitable image with the required software, but this can be found on Docker Hub of good quality and with a traceable origin.

The rest is quickly accomplished. If you want to extend Kubernetes to include a MariaDB instance with Galera using a stateful set, you then create the matching pod definitions as a YAML file and load them into your cluster. A few seconds later, you have a highly available, scalable database with persistent storage. What's more, Kubernetes will actively take care of database operations and replace failed pods with new ones if a piece of hardware bites the dust.

All told, this kind of setup is easier to create on a conventional Linux distribution than a legacy setup, and it offers improved functionality once



Details

Cluster API Address
https://api.ci-ln-py8w3rk-d5d6b.origin-ci-int-aws.dev.rhcloud.com:6443

Cluster ID
5d9846c2-92cb-4c8c-bfa8-68886bf3a884
OpenShift Cluster Manager

Provider
AWS

OpenShift Version
4.3.0-0.nightly-2020-01-06-185654

Update Channel
stable-4.3

Cluster Inventory

6 Nodes
191 Pods
1 Storage Class
0 PVCs

No cluster alerts or messages

Cluster Utilization

Resource	Usage	21:00	21:02	21:04	21:06
CPU 21.45 available	2.55 of 24				
Memory 80.95 GiB available	13 GiB of 93.95 GiB				
Filesystem 2.65 TiB available	202 GiB of 2.85 TiB				

Ongoing

There are no ongoing activities.

Recent Events

21:07 Updated machine ci-ln-py...
21:07 Updated machine ci-ln-py...
21:07 Updated machine ci-ln-py...
21:07 Updated machine ci-ln-py...
21:07 Updated machine ci-ln-py...
21:07 Updated machine ci-ln-py...
20:59 Created container pruner
20:59 Started container pruner
20:59 Container image "quay.io/...
20:59 Created Pod/revision-pru...

Figure 4: Large Kubernetes distributions such as OpenShift offer massive feature sets, but in many cases they are not needed in an SME. © Red Hat

in place – and it is true not just for MariaDB. After all, once Kubernetes has been set up, you have the option of running other services for which you previously used in-house hardware. What this means is that Kubernetes genuinely lets SMEs consolidate their in-house server farm. As a result, having less bare metal that needs to be maintained individually is a good thing because it saves you time.

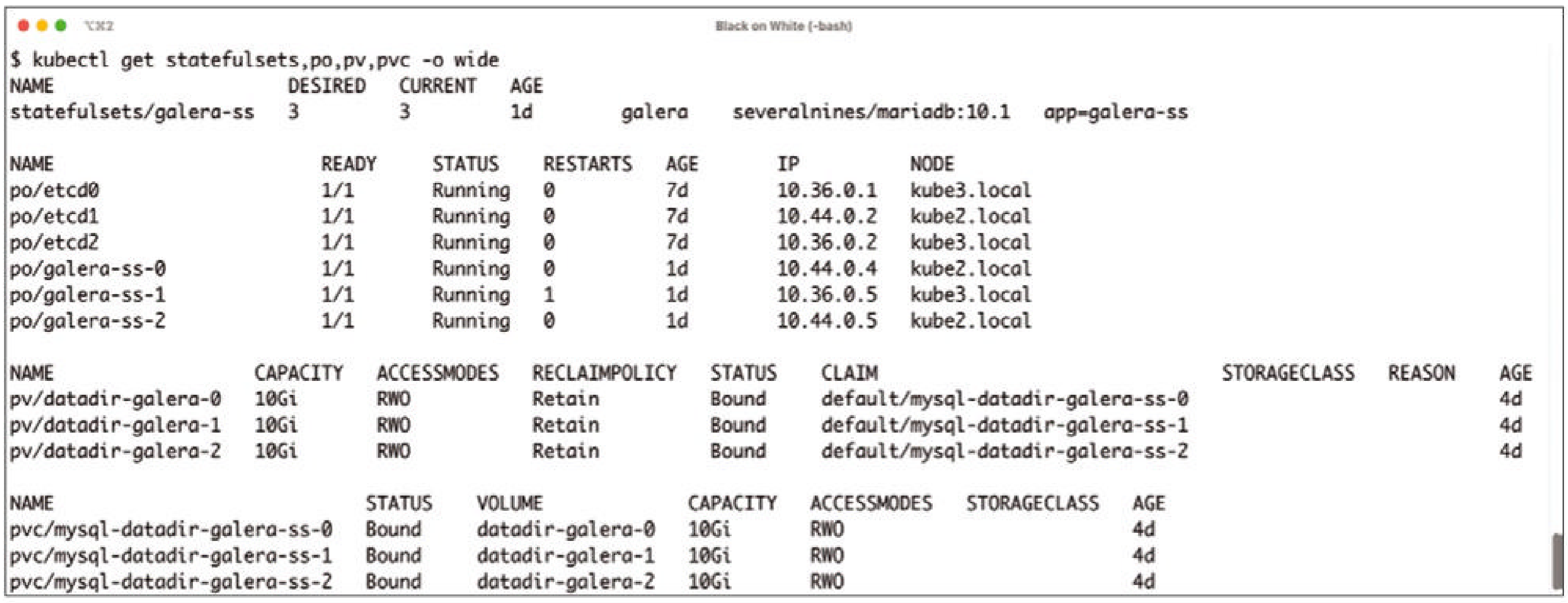
Icing on the Cake

Finally, the question arises as to whether smaller organizations can also benefit from the colorful world of cloud-native applications. Remember, cloud native means a form of application development that does away with large monoliths and instead relies on a network of microprograms that communicate with each other over defined interfaces. Each mini-app has a single, specific task.

The model has many advantages in practice. For example, it means that the individual components can be developed completely independently of each other and do not require common release cycles. It also implicitly avoids the horror scenario of release day fails, where a huge, monolithic application finally finds its way into production after months of development. Often enough, companies only discover what is not working at this point and frantically beat a retreat. The cloud-native development model avoids this issue. That said, many SMEs do not use in-house applications that would have these issues. This situation is all the more true if the company's focus is not on IT, but IT is a means to an end. If you do not develop applications in-house, you will not benefit from the cloud-native principle initially. On the other hand, if you have existing applications with an old design, you will certainly benefit from migrating to a cloud-native approach. Many large companies have shown how this can be done by consistently ditching old habits and, in some cases, discarding applications that have been used for decades and rebuilding them on cloud-native lines. Of course, this presupposes the luxury of having the time and resources to do so. For many SMEs, this is probably not the case.

Cloud-Native Help

From an admin's point of view, it might be worthwhile to take a closer



```
$ kubectl get statefulsets,po,pv,pvc -o wide
NAME                   DESIRED   CURRENT   AGE
statefulsets/galera-ss   3         3         1d        galera     severalnines/mariadb:10.1   app=galera-ss

NAME                     READY     STATUS    RESTARTS   AGE      IP            NODE
po/etcd0                 1/1       Running   0          7d       10.36.0.1     kube3.local
po/etcd1                 1/1       Running   0          7d       10.44.0.2     kube2.local
po/etcd2                 1/1       Running   0          7d       10.36.0.2     kube3.local
po/galera-ss-0           1/1       Running   0          1d       10.44.0.4     kube2.local
po/galera-ss-1           1/1       Running   1          1d       10.36.0.5     kube3.local
po/galera-ss-2           1/1       Running   0          1d       10.44.0.5     kube2.local

NAME                  CAPACITY   ACCESSMODES   RECLAIMPOLICY   STATUS    CLAIM                              STORAGECLASS   REASON    AGE
pv/datadir-galera-0   10Gi       RWO           Retain          Bound     default/mysql-datadir-galera-ss-0                              4d
pv/datadir-galera-1   10Gi       RWO           Retain          Bound     default/mysql-datadir-galera-ss-1                              4d
pv/datadir-galera-2   10Gi       RWO           Retain          Bound     default/mysql-datadir-galera-ss-2                              4d

NAME                          STATUS    VOLUME             CAPACITY   ACCESSMODES   STORAGECLASS   AGE
pvc/mysql-datadir-galera-ss-0   Bound     datadir-galera-0   10Gi       RWO                          4d
pvc/mysql-datadir-galera-ss-1   Bound     datadir-galera-1   10Gi       RWO                          4d
pvc/mysql-datadir-galera-ss-2   Bound     datadir-galera-2   10Gi       RWO                          4d
```

Figure 5: With a stateful set, Galera can easily be run in Kubernetes with a great deal of automation, offering quite significant advantages over a legacy approach.

look at some of the principles of cloud-native design. Even if your own application is a containerized monolith, the cloud-native approach has brought about a number of changes that make sense outside of native cloud apps.

One good example of this is sidecars, which can be installed in Kubernetes parallel to the containers with the applications. In the above example with MariaDB and Galera, one sidecar would definitely add some value.

The one I have in mind is Istio (Figure 6), a tool that controls the connections between the individual components in micro-apps, encrypts them, and acts as a load balancer. However, even a Galera cluster needs a load balancer to offer meaningful access to clients because you can't give most database clients multiple IP addresses. On top of that, doing so would be clumsy because – in the worst case – the client might just connect to a Galera instance that is not working.

Kubernetes only offers a simple balancer in the form of a service IP, whereas Istio can do far more and can be rolled out as a sidecar to Galera. Because Istio runs as a highly available component in Kubernetes, you can even save yourself the trouble of running a separate load balancer, regardless of whether it uses Linux or comes in the form of an appliance.

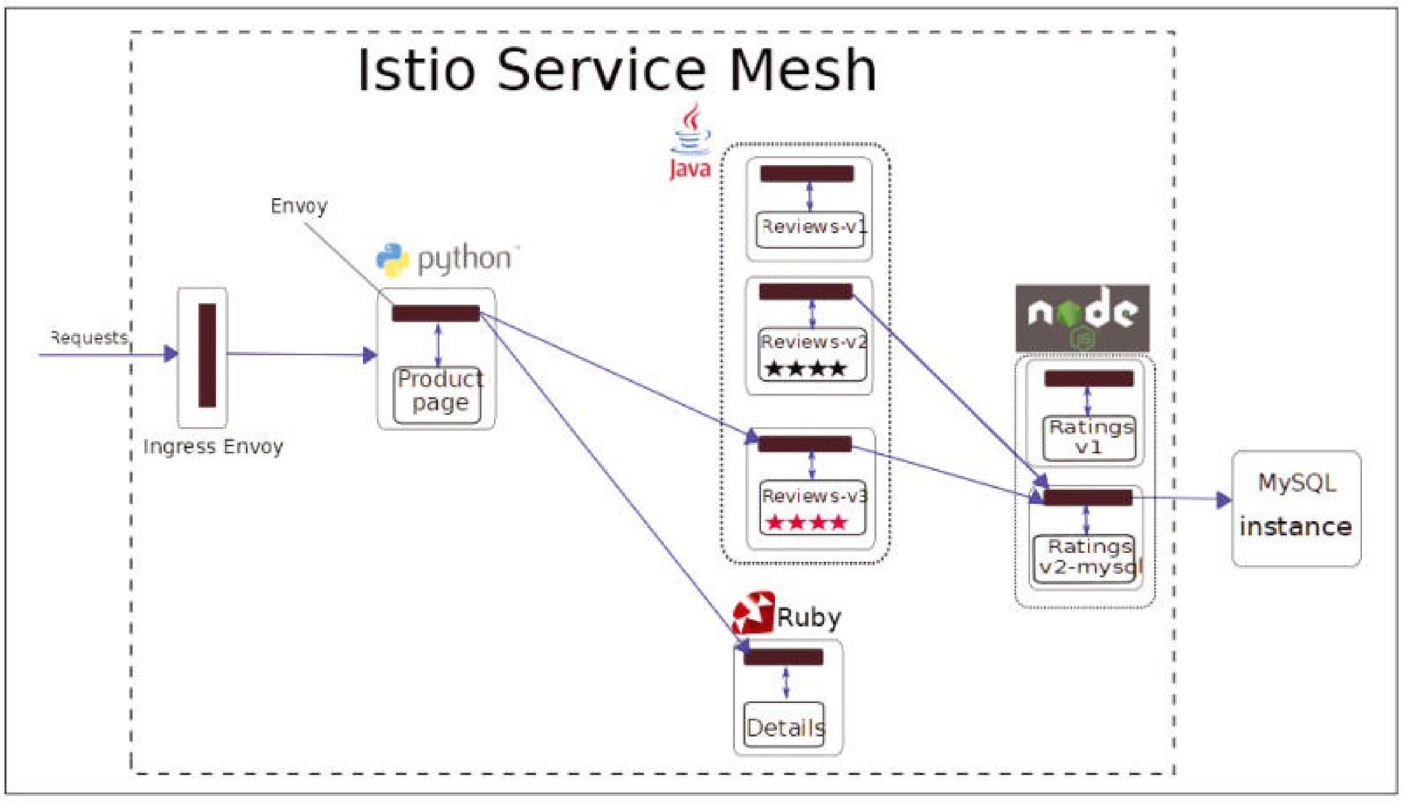


Figure 6: Although Istio is intended for the world of cloud-native applications, it can also be used meaningfully with conventional services such as MySQL or Galera. © Istio

Words of Warning

Despite all the euphoria about container orchestration and the added value that containers undoubtedly offer, even for small companies, a warning will not go amiss at the end of this article. Many have already proclaimed this warning several times, but the problem has hardly changed in the meantime.

The idea of shipping an application and its userland as a container has many merits, but in a corporate environment, you should only use vendor-provided images for which you have no doubt about how they were built and only if you can be sure that the vendor regularly maintains its images.

If this is not the case, it makes more sense in most cases to bite the bullet and bundle the application in a container of your own design. A CI/CD system for this can be implemented without too much hardware, as mentioned earlier. The reward for this extra work is the certainty that, as an admin, you can respond quickly to problems such as vulnerabilities without having to rely on a provider.

Customers of vendors that offer their applications as black box containers also need to communicate clearly to the vendor that this is unacceptable. In your own interest, you should only use containers in production if you are familiar with and understand their inner workings.

Conclusions

The examples presented in this article are precisely that – examples that offer only a tiny glimpse into what containers and Kubernetes can do for SMEs. Clearly, however, massive corporations are not the only ones who can benefit from containerization. Kubernetes is easier to operate and maintain than conventional setups, it comes with masses of automation, and it scales better across the board than its predecessors.

Even if you are not comfortable with Kubernetes, you will find many good approaches to making your own installation easier to maintain and more efficient in the trend toward containerization. Both containers and Kubernetes ultimately help SMEs boost the level of automation, which can be more important in small companies than in large corporations, not least because IT is often just a sideline for which too few personnel are available. The rule is: The more automation, the better. Containers and Kubernetes make a genuine contribution toward this end. ■

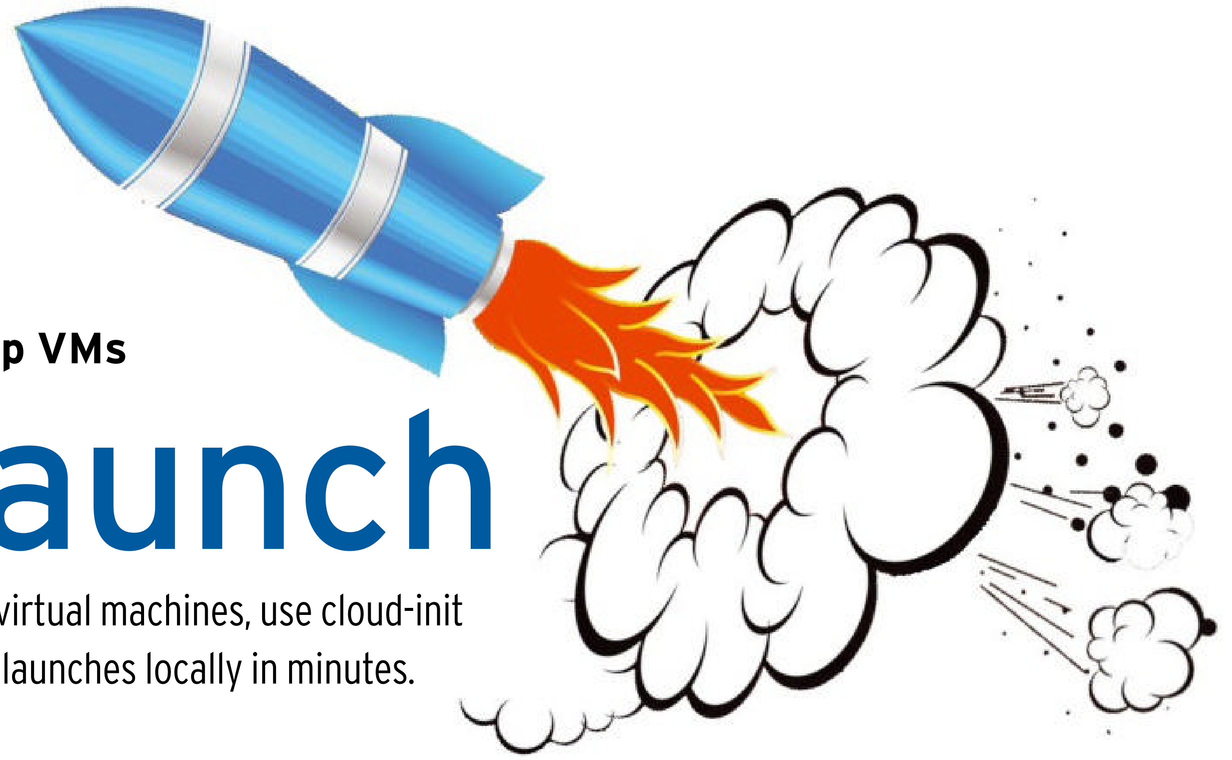
Info

[1] IDG study on cloud-native applications: [https://www.suse.com/c/idg-study-cloud-native-2022-where-do-european-companies-stand-in-their-digital-transformation/]

[2] Microk8s: [https://microk8s.io]

[3] K3s: [https://k3s.io]

[4] Minikube: [https://kubernetes.io/blog/2019/03/28/running-kubernetes-locally-on-linux-with-minikube-now-with-kubernetes-1.14-support/]



Goodbye cloud VMs, hello laptop VMs

Local Launch

Multipass lets you launch and run Ubuntu virtual machines, use cloud-init to configure the VMs, and prototype cloud launches locally in minutes.

By Ankur Kumar

In a previous article [1], I explored how to bring up cloud-like virtual machine stacks on a laptop with container machine tooling. That approach didn't require a lot of computing resources or any hardware-based virtualization support. However, security is a downside of containers because they provide thinner isolation compared with the strong isolation provided by virtual machines. Does a lightweight and fast solution exist to create cloud VM types of stacks on a laptop? Also desirable is tooling that works out of the box without a lot of dependencies to gather, install, and so on. The answer is "Yes." You can have your cake and eat it too with Multipass [2]!

The tooling provided in this article is for newer machines that have virtualization capability provided by the processor. Still, the container machine approach presented in the previous article is an option, in case you don't have such hardware available. I tested the snippets shown or referenced in this article on my four-core, 8GB COREi7 laptop running Ubuntu 18.04 LTS, but I have also used the tool on macOS in a professional setting.

CloudVMs Through Multipass

Multipass [3] is a tool to bring up Ubuntu virtual machines that, like public cloud VMs, can be configured by `cloud-init`. The tool is a lightweight, cross-platform, virtual machine manager (VMM) that uses the standard underlying hypervisors on the respective platform to keep the overhead minimal. With Multipass, you can enact a public-cloud-like VM environment anywhere on demand.

Getting Started

To begin, install the latest stable version Multipass on an Ubuntu machine through snap mechanism:

```
sudo snap install multipass
```

You can also see the available channels and respective versions (e.g., `candidate/beta/edge`) and install one of the alternatives:

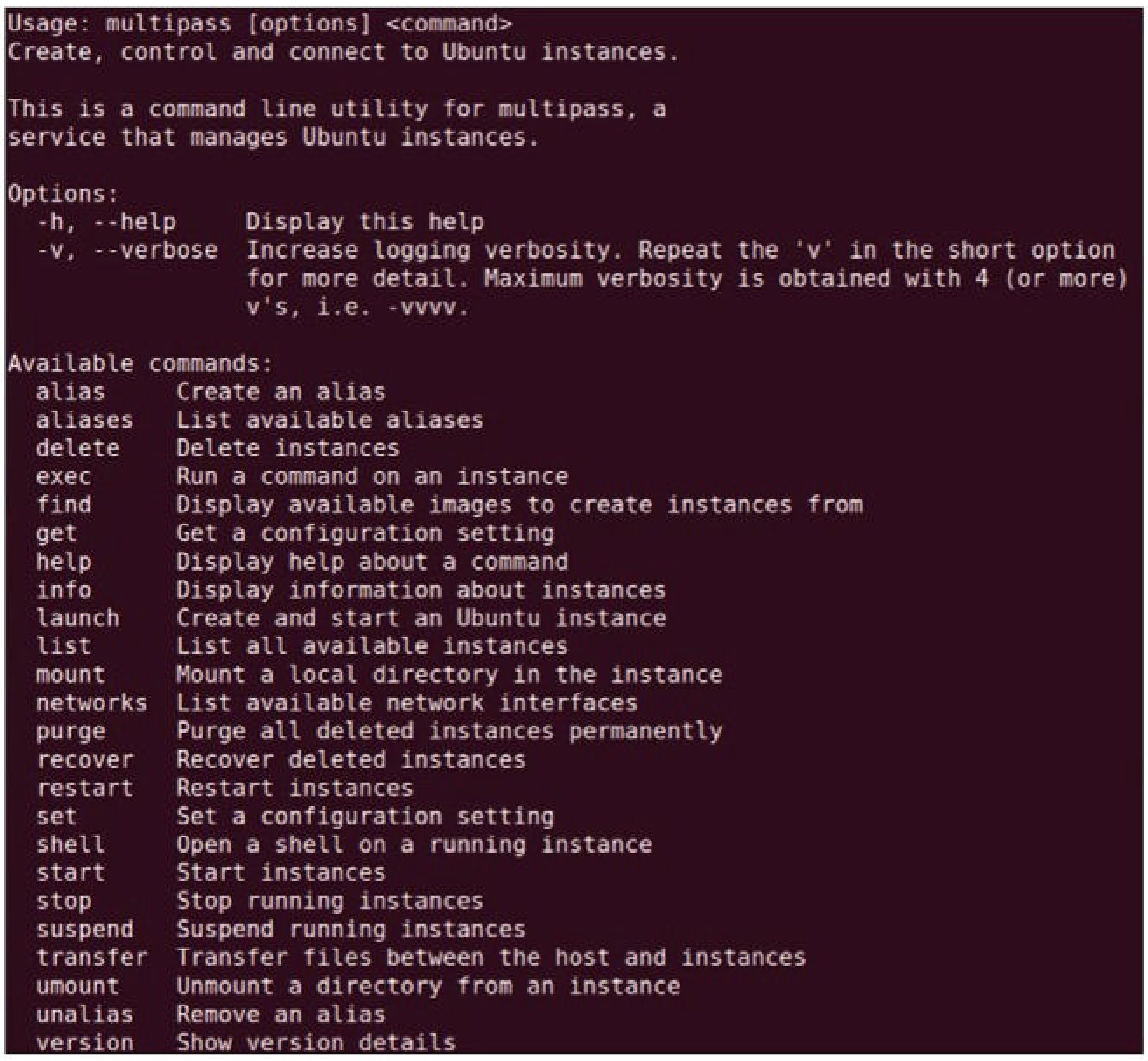
```
snap info multipass
sudo snap refresh multipass <--stable|--candidate|--beta|--edge>
```

Executing the `multipass` command in your terminal should dump a help screen (Figure 1).

You need to find an appropriate Ubuntu cloud image to instantiate your first VM. The command

```
multipass find
```

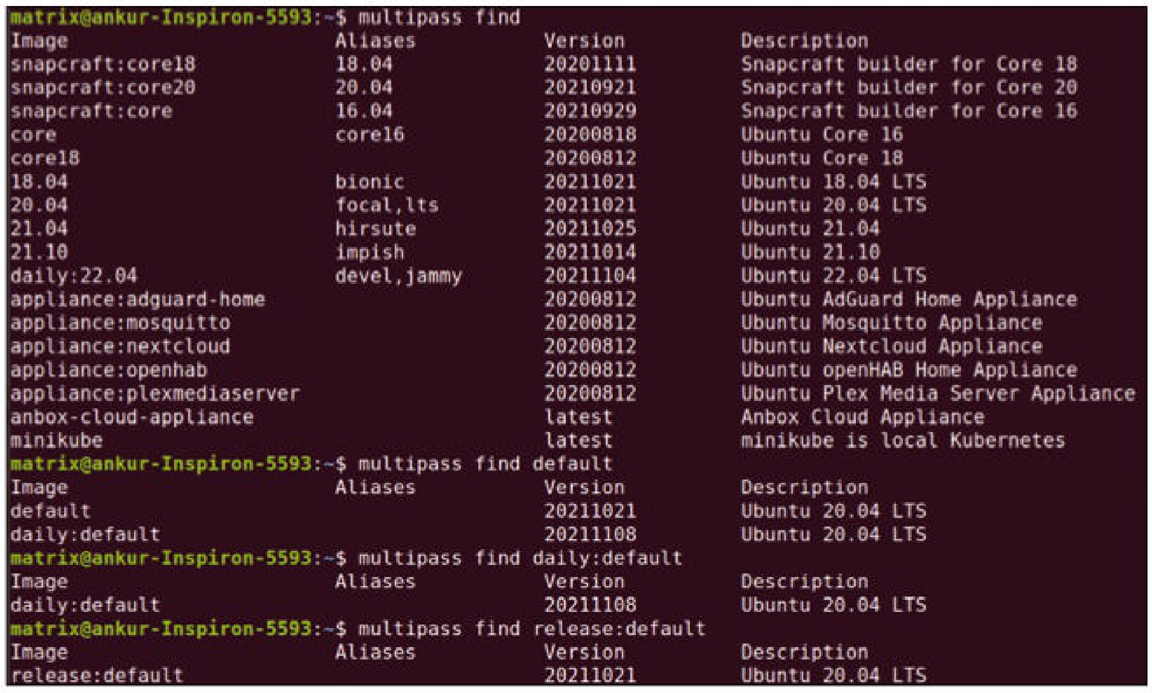
dumps all available and supported images (Figure 2). You could also append the `--unsupported` option, if you want to experiment with old



```
Usage: multipass [options] <command>
Create, control and connect to Ubuntu instances.

This is a command line utility for multipass, a
service that manages Ubuntu instances.

Options:
  -h, --help     Display this help
  -v, --verbose  Increase logging verbosity. Repeat the 'v' in the short option
                 for more detail. Maximum verbosity is obtained with 4 (or more)
                 v's, i.e. -vvvv.

Available commands:
  alias     Create an alias
  aliases   List available aliases
  delete    Delete instances
  exec      Run a command on an instance
  find      Display available images to create instances from
  get       Get a configuration setting
  help      Display help about a command
  info      Display information about instances
  launch    Create and start an Ubuntu instance
  list      List all available instances
  mount     Mount a local directory in the instance
  networks  List available network interfaces
  purge     Purge all deleted instances permanently
  recover   Recover deleted instances
  restart   Restart instances
  set       Set a configuration setting
  shell     Open a shell on a running instance
  start     Start instances
  stop      Stop running instances
  suspend   Suspend running instances
  transfer  Transfer files between the host and instances
  umount    Unmount a directory from an instance
  unalias   Remove an alias
  version   Show version details
```

Figure 1: Multipass help screen.

Lead Image © Liubomyr Feshchyn, 123rf.com



```
matrix@ankur-Inspiron-5593:~$ multipass find
Image                       Aliases           Version          Description
snapcraft:core18            18.04             20201111         Snapcraft builder for Core 18
snapcraft:core20            20.04             20210921         Snapcraft builder for Core 20
snapcraft:core              16.04             20210929         Snapcraft builder for Core 16
core                        core16            20200818         Ubuntu Core 16
core18                                        20200812         Ubuntu Core 18
18.04                       bionic            20211021         Ubuntu 18.04 LTS
20.04                       focal,lts         20211021         Ubuntu 20.04 LTS
21.04                       hirsute           20211025         Ubuntu 21.04
21.10                       impish            20211014         Ubuntu 21.10
daily:22.04                 devel,jammy       20211104         Ubuntu 22.04 LTS
appliance:adguard-home                        20200812         Ubuntu AdGuard Home Appliance
appliance:mosquitto                           20200812         Ubuntu Mosquitto Appliance
appliance:nextcloud                           20200812         Ubuntu Nextcloud Appliance
appliance:openhab                             20200812         Ubuntu openHAB Home Appliance
appliance:plexmediaserver                     20200812         Ubuntu Plex Media Server Appliance
anbox-cloud-appliance                         latest           Anbox Cloud Appliance
minikube                                      latest           minikube is local Kubernetes
matrix@ankur-Inspiron-5593:~$ multipass find default
Image                       Aliases           Version          Description
default                                       20211021         Ubuntu 20.04 LTS
daily:default                                 20211108         Ubuntu 20.04 LTS
matrix@ankur-Inspiron-5593:~$ multipass find daily:default
Image                       Aliases           Version          Description
daily:default                                 20211108         Ubuntu 20.04 LTS
matrix@ankur-Inspiron-5593:~$ multipass find release:default
Image                       Aliases           Version          Description
release:default                               20211021         Ubuntu 20.04 LTS
```

Figure 2: Multipass `find` output.

unsupported versions of Ubuntu, or filter the images by appending a `<remote>`, which is either `release:` or `daily:`. To get information about the default image, include `default`, with or without a remote.

Local Cloud VM

To bring up a local Ubuntu VM with the default image, enter

```
multipass launch
```

Appending a desired image or alias to the command instantiates the VM with that particular image. Table 1 shows the various arguments you can append to the `launch` command. For example, the command

```
multipass launch -n mycloudvm -c 2 ↩
  -m 2G -d 10G --timeout 600
```

brings up a local VM with the default LTS release version of Ubuntu (20.04 as of this writing) named *mycloudvm* with two CPUs, 2GB of memory, 10GB of disk space, and a launch operation timeout of 10min.
The commands

```
multipass list
multipass info --all
```

dump all information about the launched instances. The `info` command adds some extra runtime details. This information could be dumped in other than the default format (a table for v1.8.0) by appending `--format <json|csv|yaml>` to the `list` or `info` command. Once your instance is up, use

```
multipass shell <VMName>
```

to log in. Now your local cloud VM instance is ready to be used as per your desire (Figure 3).
If you don't provide a name for the VM, an instance named *primary* with default resources is created (if it does not already exist). The *primary* instance automounts your home directory, but you could unmount that instance with the command:

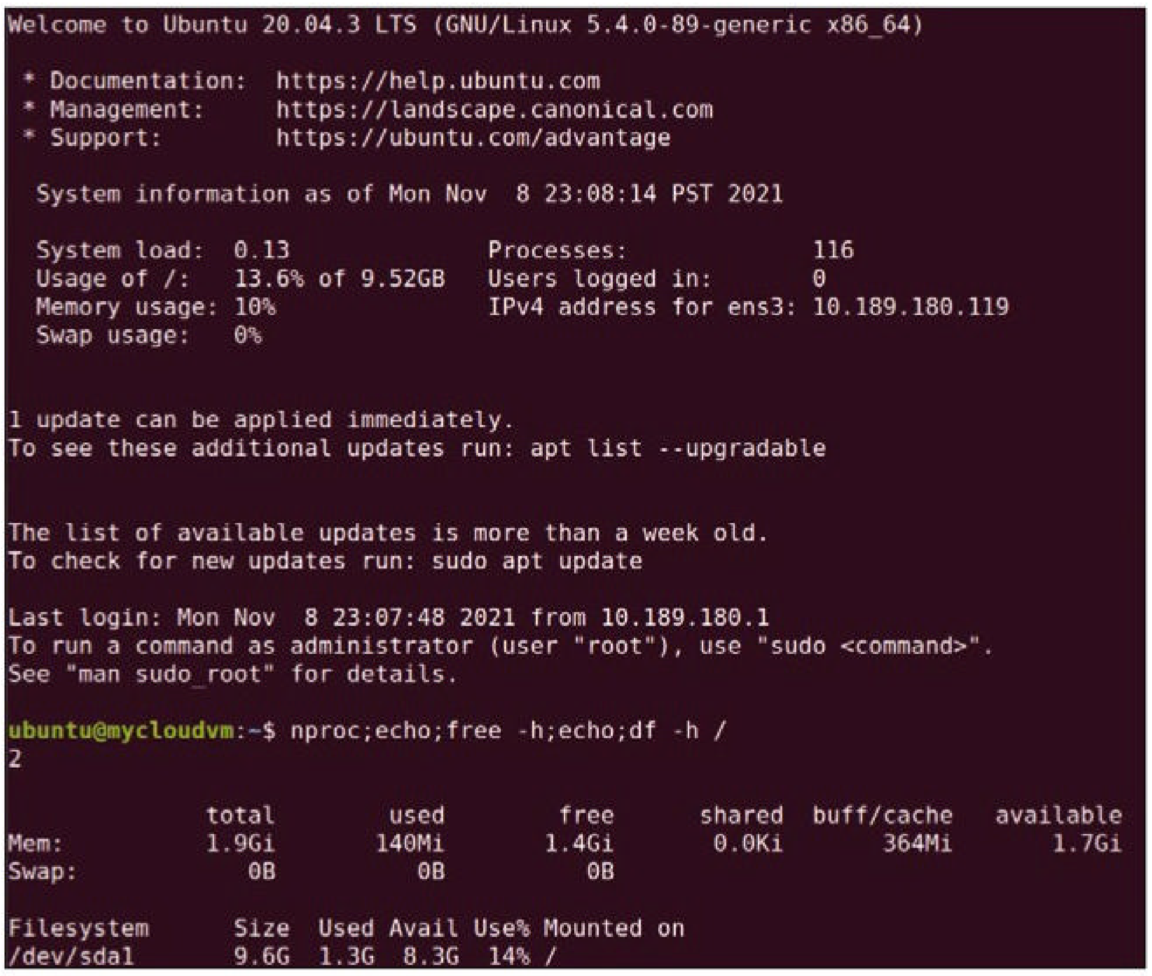
```
multipass unmount primary
```

If you want to set the name of the primary instance to something other than the default, you can use the command:

```
multipass set client.primary-name=↩
  <CustomPrimaryName>
```

Table 1: Multipass `launch`

Argument	Action (v1.8.0 default)
`-n <NAME>`	Override the auto-generated name with a name of your choice (random name).
`-c <NumOfCPUs>`	Specify the number of CPUs (1).
`-m <AmountOfMemory>`	Specify memory to allocate (1GB).
`-d <DiskSpace>`	Specify storage space (5GB).
`--timeout <TimeoutInSecs>`	Maximum time in seconds to bring up the VM instance (max 5min).



```
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-89-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Nov  8 23:08:14 PST 2021

  System load:  0.13              Processes:               116
  Usage of /:   13.6% of 9.52GB   Users logged in:         0
  Memory usage: 10%               IPv4 address for ens3: 10.189.180.119
  Swap usage:   0%


1 update can be applied immediately.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Mon Nov  8 23:07:48 2021 from 10.189.180.1
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@mycloudvm:~$ nproc;echo;free -h;echo;df -h /
2

              total        used        free      shared  buff/cache   available
Mem:          1.9Gi       140Mi       1.4Gi       0.0Ki       364Mi       1.7Gi
Swap:            0B          0B          0B

Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1       9.6G  1.3G  8.3G  14% /
```

Figure 3: A local cloud VM instance created by Multipass.

Similarly, the multipass start command launches a primary instance if one has not been created already.

To execute any shell command on a running instance, use multipass exec. Logically, you could combine transfer and exec to copy and execute your configuration scripts and create automation around the instances.

The multipass commands stop, start, restart, suspend, and delete are self-explanatory. You can perform those operations on named instances or use --all for all instances.

The multipass recover command is for recovering deleted instances, and purge cleans up after the deleted instances. The multipass commands take some other options, too, and you could dump the detailed info about any command with multipass help <Command>.

Advanced Tooling with Multipass

Multipass can use cloud-init to configure local VMs fully, as used in various public clouds. With this tool, you can prototype your public cloud launches locally, for free, and in a way that can not be accomplished with similar tools. The cloud-init tool initializes VM instances – fully configured machines out of the box – without the need for an external configuration tool. For example, I use this functionality to bring up fully configured single instances running many standard cloud components on my laptop, allowing quick experiment-verify-cleanup combinations of various stacks, sometimes a few dozen times a day, before the final public cloud push.

Listing 1 shows a cloud-init configuration in the cloud-config data format with its various required sections. I use this file with Multipass to bring up a fully configured instance running a Dockerized Apache Cassandra and its web user interface (UI) out of the box. The users section configures a user named ubuntu with sudo privileges, ready to use the desired SSH keypair for the SSH login.

Listing 1: cloud-config-cassandra.yaml

```
#cloud-config
users:
  - default
  - name: ubuntu
    gecos: Ubuntu
    sudo: ALL=(ALL) NOPASSWD:ALL
    groups: users, admin
    shell: /bin/bash
    ssh_import_id: None
    lock_passwd: true
    ssh_authorized_keys:
      - PUBLICKEY
package_update: true
package_upgrade: true
packages:
  - avahi-daemon
write_files:
  - content: |
      version: "2.4"
      services:
        cassandra:
          image: cassandra:4.0
          container_name: cassandra
          hostname: cassandra
          mem_limit: 1.8g
          healthcheck:
            test: ["CMD", "cqlsh", "-e", "describe keyspaces"]
            interval: 5s
            timeout: 5s
            retries: 10
          network_mode: host
          volumes:
            - cassandra-data:/var/lib/cassandra
          restart: unless-stopped
          environment:
            CASSANDRA_CLUSTER_NAME: CassandraTest
            CASSANDRA_DC: CassandraTest
            CASSANDRA_ENDPOINT_SNITCH: GossipingPropertyFileSnitch
            CASSANDRA_NUM_TOKENS: 4
            CASSANDRA_RACK: CassandraTest
        cassandrawu:
          image: ipushc/cassandra-web
          container_name: cassandrawu
          hostname: cassandrawu
          mem_limit: 512m
          network_mode: host
          restart: unless-stopped
          environment:
            HOST_PORT: ":8080"
            CASSANDRA_HOST: localhost
          depends_on:
            cassandra:
              condition: service_healthy
      volumes:
        cassandra-data:
    path: /opt/docker/compose/cassandra.yml
    owner: ubuntu:ubuntu
    permissions: '0600'
runcmd:
  - wget "https://get.docker.com" -O get-docker.sh
  - sh get-docker.sh
  - rm -fv get-docker.sh
  - usermod -aG docker ubuntu
  - wget "https://github.com/docker/compose/releases/download/1.29.2/
           docker-compose-Linux-x86_64" -O /usr/local/bin/docker-compose
  - chmod 0755 /usr/local/bin/docker-compose
  - mkdir -p /opt/docker/compose
  - docker-compose -f /opt/docker/compose/cassandra.yml up -d
output:
  init:
    output: "> /var/log/cloud-init.out"
    error: "> /var/log/cloud-init.err"
  config: "tee -a /var/log/cloud-config.log"
  final:
    - ">> /var/log/cloud-final.out"
    - "/var/log/cloud-final.err"
```

Note that `PUBLICKEY` is a placeholder, and you need to replace it with the public key content of the key pair. If you don't have an SSH key pair generated, use `ssh-keygen` and follow the prompts. The `package_update` section resynchronizes the package index files from their sources, and the `package_upgrade` section installs the newest versions of all packages currently installed on the system. The `packages` section takes a list of the packages that need to be installed on the system.

The example configuration installs the `avahi-daemon` to resolve `<hostname>.local` to the instance address.

The `write_files` section generates files by combining content, path, owner, and permission for each file. The example configuration generates a docker-compose manifest to bring up Cassandra and its web UI stack. The `runcmd` section takes a list of the shell commands to run in the order given to complete the configuration of the instance. Finally, the `output` section generates various logs capturing stdout and stderr during various stages of `cloud-init` to debug in case the instance is not configured as per the `cloud-config` data provided. The `cloud-init` service and `cloud-config` script provide more functionality through many other sections. Please refer to the `cloud-config` examples [4] in the reference section to see those in action.

To bring up a fully configured cloud VM instance locally, just execute:

```
multipass launch -m 2G ↩
  -n cassandra --cloud-init ↩
  ./cloud-config-cassandra.yaml
```

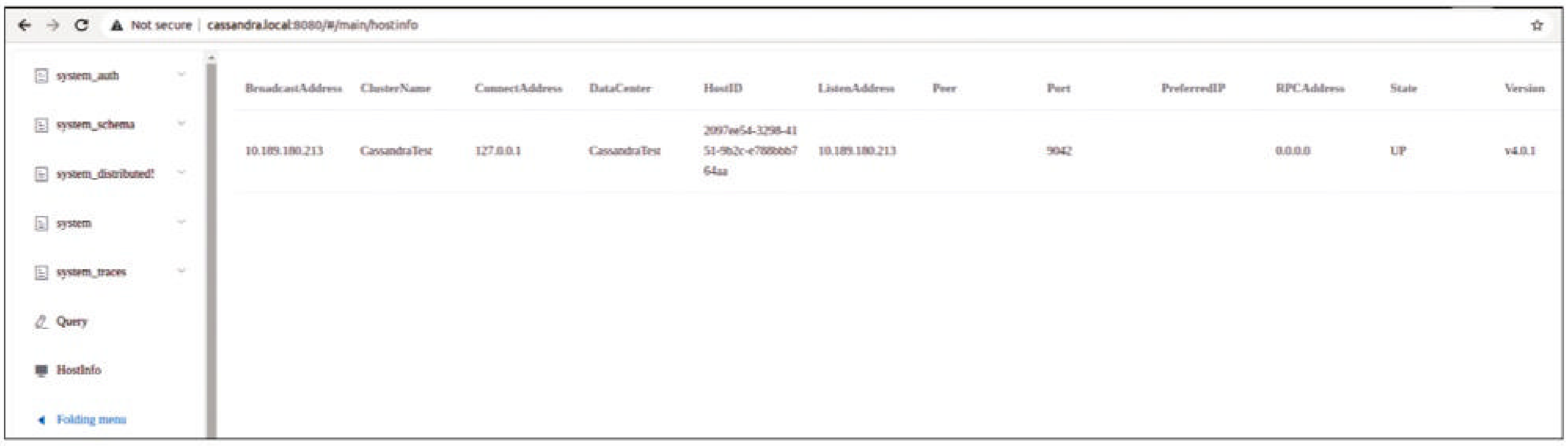
Once the instance is up, try to access the Cassandra web UI by typing *cassandra.local:8080* in your browser (Figures 4 and 5).

Log in to the Cassandra instance with command

```
ssh -oStrictHostKeyChecking=↩
  no ubuntu@cassandra.local
```

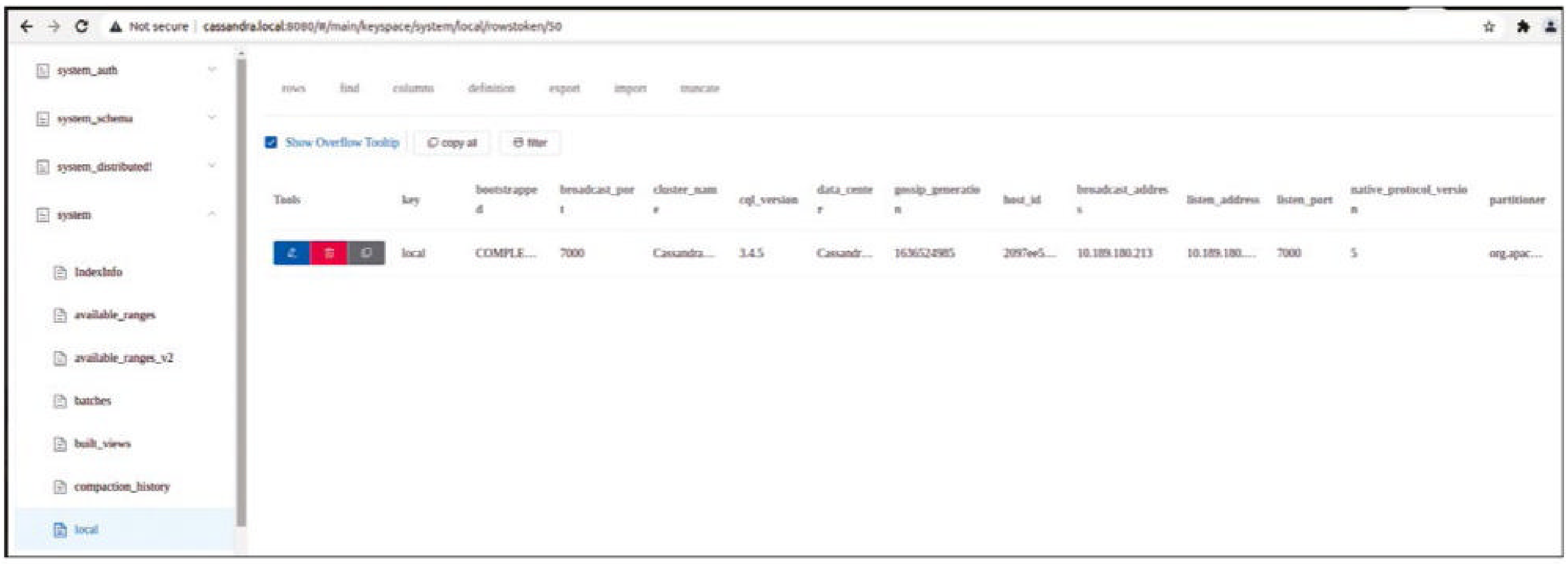
and execute the command

```
multipass delete cassandra && ↩
  multipass purge && ↩
  ssh-keygen -R cassandra.local
```



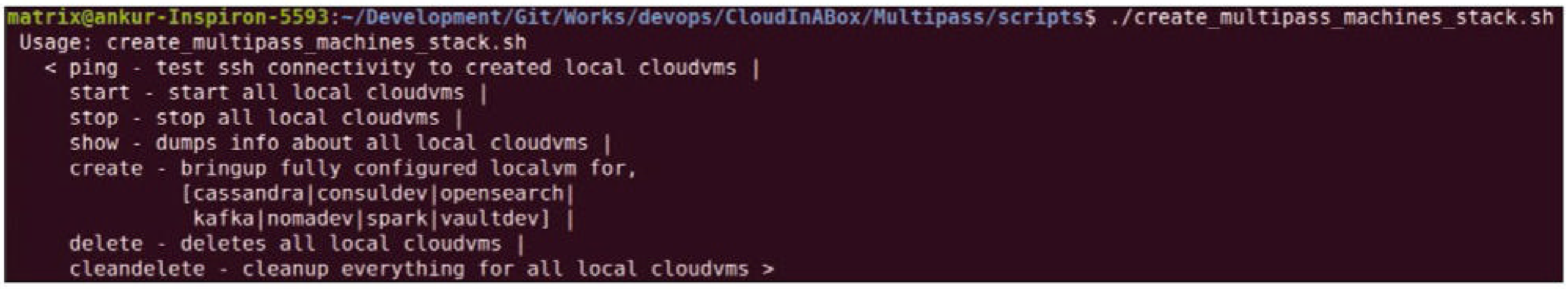
BroadcastAddress	ClusterName	ConnectAddress	DataCenter	HostID	ListenAddress	Peer	Port	PreferredIP	RPCAddress	State	Version
10.189.180.213	CassandraTest	127.0.0.1	CassandraTest	2097ee54-3298-4151-9b2c-e788bbb764aa	10.189.180.213		9042		0.0.0.0	UP	v4.0.1

Figure 4: Cassandra host info.



Tools	key	bootstrapped	broadcast_port	cluster_name	cql_version	data_center	gossip_generation	host_id	broadcast_address	listen_address	listen_port	native_protocol_version	partitioner
	local	COMPLE...	7000	Cassandra...	3.4.5	Cassandr...	1636524985	2097ee5...	10.189.180.213	10.189.180....	7000	5	org.apac...

Figure 5: Cassandra system local key



```
matrix@ankur-Inspiron-5593:~/Development/Git/Works/devops/CloudInABox/Multipass/scripts$ ./create_multipass_machines_stack.sh
 Usage: create_multipass_machines_stack.sh
   < ping - test ssh connectivity to created local cloudvms |
     start - start all local cloudvms |
     stop - stop all local cloudvms |
     show - dumps info about all local cloudvms |
     create - bringup fully configured localvm for,
              [cassandra|consuldev|opensearch|
               kafka|nomadev|spark|vaultdev] |
     delete - deletes all local cloudvms |
     cleandelete - cleanup everything for all local cloudvms >
```

Figure 6: Wrapper help screen.

to clean up when you are done with the local cloud VM.

I've created a wrapper script [5] and various cloud configs to bring up fully configured local cloud VMs on Ubuntu 20.04 LTS running Cassandra, Consul, OpenSearch, Kafka, Nomad, Spark, and Vault with a single command. To get the wrapper, just download the sources with:

```
git clone https://github.com/➘
  richnusgeeks/devops.git
pushd CloudInABox/Multipass/scripts
```

You could use cd instead of pushd, but I prefer the latter for its intelligence. The wrapper script execution should display a help screen (Figure 6).

All the local VM instances brought up with the wrapper script are configured with monit, docker, and docker-compose, in addition to the chosen target role. For example, you can bring up a fully configued local cloud VM with Consul in development mode with:

```
./create_multipass_machines_stack.sh ➘
  create consuldev
```

Once the local cloud VM instance is up, you can access the Monit (Figure 7) and Consul (Figure 8) web UIs on *consuldev.localhost:2812* (credentials: *guest/guest*) and *consuldev.localhost:8500*, respectively. When done, you can get into the created instance and clean up everything with:

```
ssh -oStrictHostKeyChecking=➘
  no ubuntu@consuldev.local
./create_multipass_machines_stack.sh ➘
  cleandelete
```

The tooling with the wrapper script around Multipass made bringing up and cleaning up the fully configured local cloud VMs easy.

Conclusion

Multipass is a must-have tool for modern cloud and infra/platform developers. Combined with cloud-config data, it is highly useful for bringing up almost fully configured Ubuntu public cloud instances locally. With Multipass, you can accelerate productivity while slashing your public cloud bill. ■

Info

[1] "Footloose" by Ankur Kumar, *ADMIN*, issue 68, 2022, pg. 46, [https://www.admin-magazine.com/Archive/2022/68/Goodbye-virtual-machines-hello-container-machines]

[2] Multipass homepage: [https://multipass.run/]

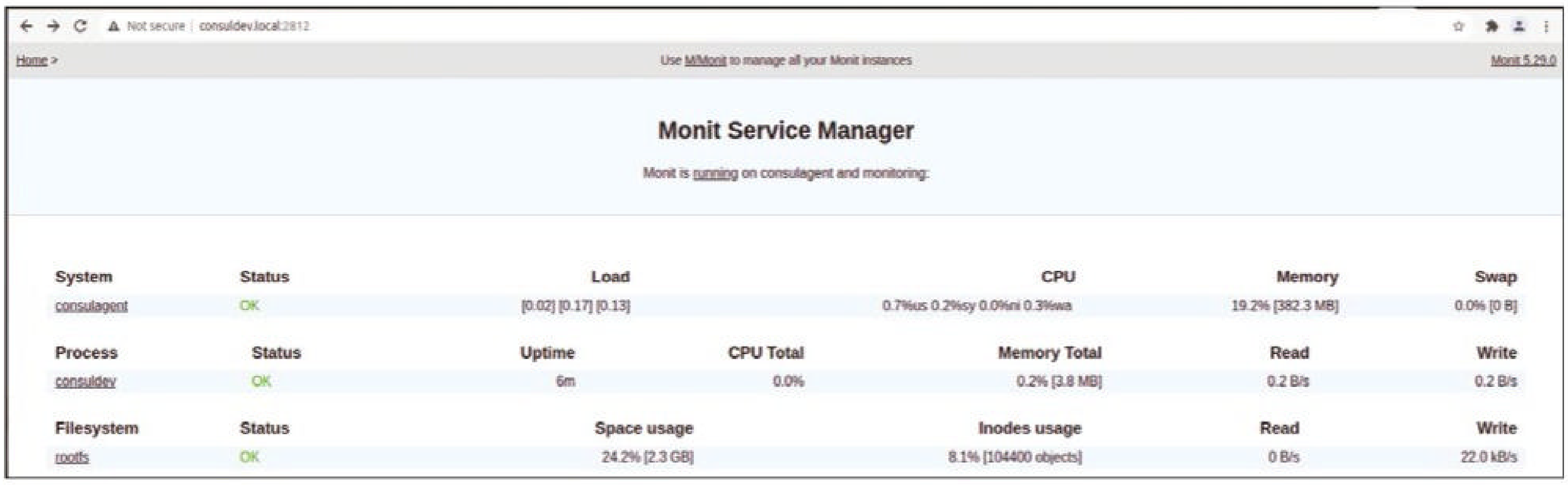
[3] Multipass on GitHub: [https://github.com/canonical/multipass]

[4] Cloud-config examples: [https://cloudinit.readthedocs.io/en/latest/topics/examples.html#yaml-examples]

[5] Wrapper on GitHub: [https://github.com/richnusgeeks/devops/tree/master/CloudInABox/Multipass/scripts]

Author

Ankur Kumar is a passionate free and open source hacker/researcher and seeker of mystical life knowledge. He explores cutting-edge technologies, ancient sciences, quantum spirituality, and various genres of music, mystical literature, and art. You can connect with Ankur at [https://www.linkedin.com/in/richnusgeeks] and explore his GitHub site ([https://github.com/richnusgeeks]) for other useful FOSS ideas.


Not secure | consuldev.local:2812
Home >
Use M/Monit to manage all your Monit instances
Monit 5.29.0

Monit Service Manager

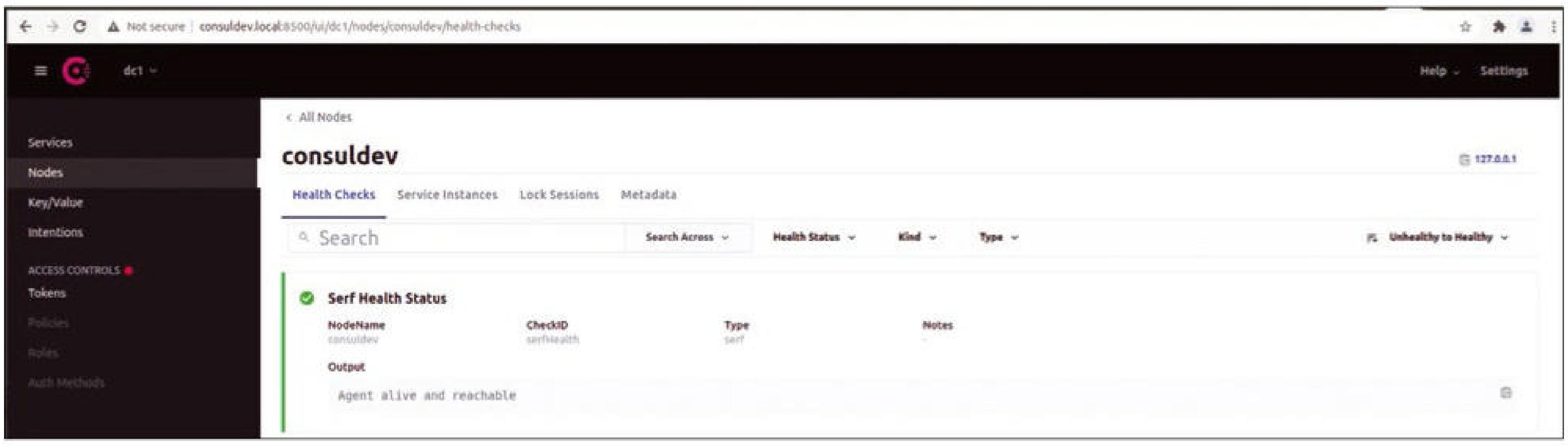
Monit is running on consulagent and monitoring:

System	Status	Load	CPU	Memory	Swap
consulagent	OK	[0.02] [0.17] [0.13]	0.7%us 0.2%sy 0.0%ni 0.3%wa	19.2% [382.3 MB]	0.0% [0 B]

Process	Status	Uptime	CPU Total	Memory Total	Read	Write
consuldev	OK	6m	0.0%	0.2% [3.8 MB]	0.2 B/s	0.2 B/s

Filesystem	Status	Space usage	Inodes usage	Read	Write
rootfs	OK	24.2% [2.3 GB]	8.1% [104400 objects]	0 B/s	22.0 kB/s

Figure 7: Monit web user interface.



consuldev

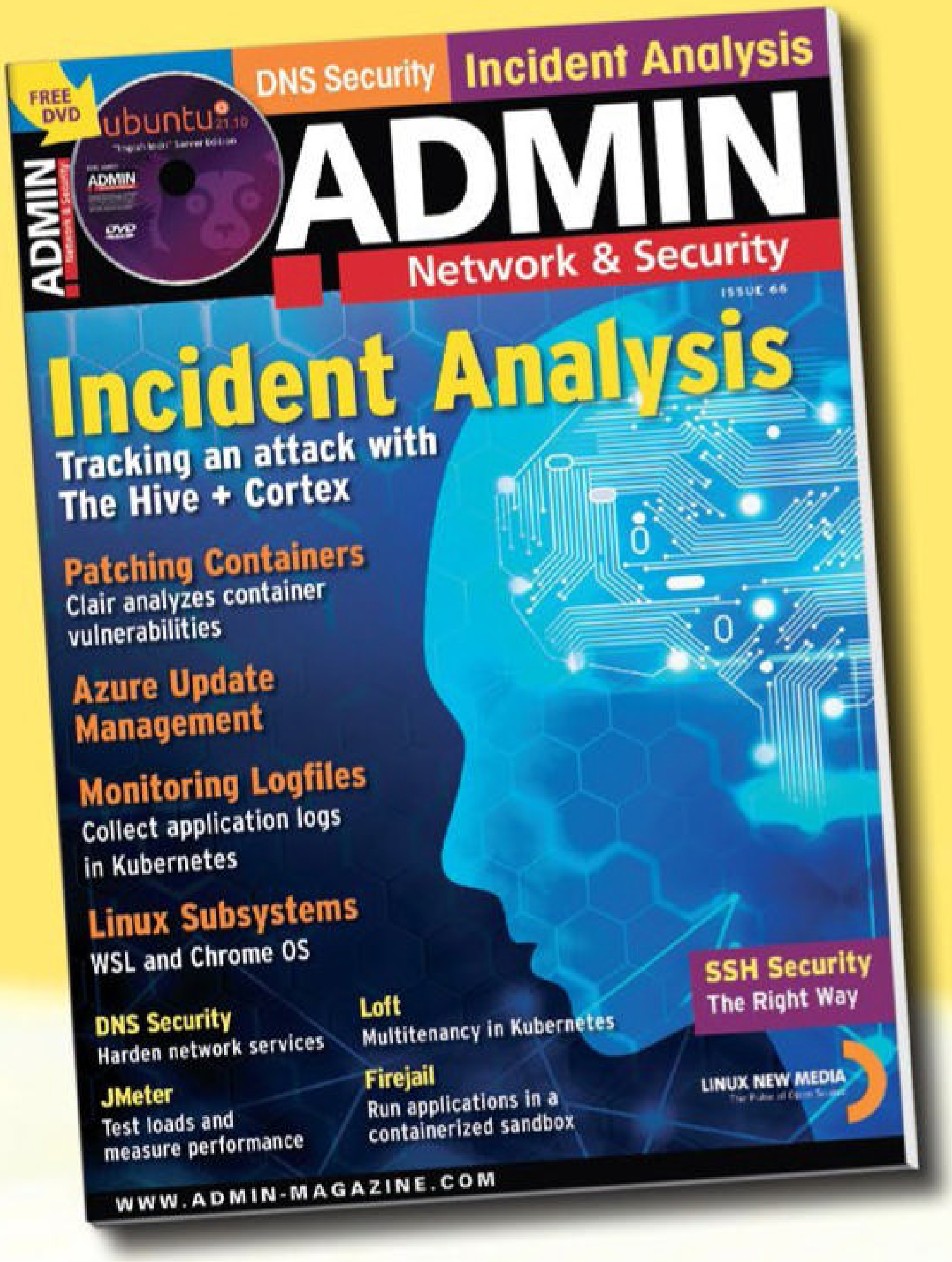
Serf Health Status

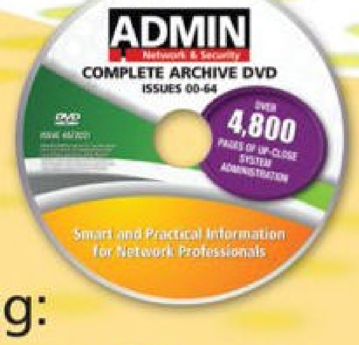
NodeName	CheckID	Type	Notes
consuldev	serfHealth	serf	-

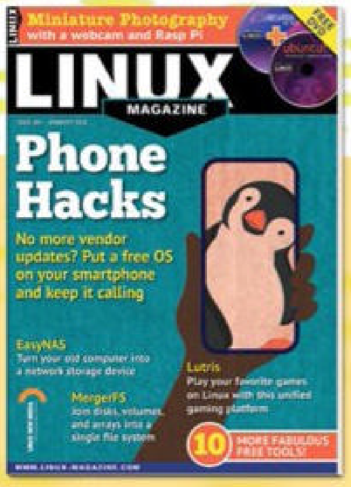
Output

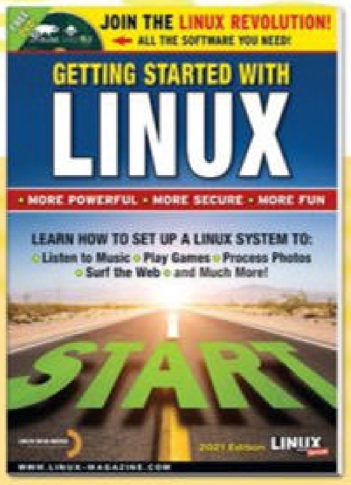
```
Agent alive and reachable
```

Figure 8: Consul web user interface.









Microsoft Exchange replacement

Top Dog in Sight

Grommunio is setting itself up as an open source drop-in replacement for Microsoft Exchange. By Markus Feilner

For decades, Microsoft Exchange has dominated the enterprise groupware market. Whether as a desktop, mobile, or plain vanilla web application, no competitor has been able to assert itself permanently against Microsoft Exchange. Possible alternatives have proved too clumsy and often lacked comparable features and scalability. Yet the market leader's product has many weaknesses, which opens a huge market opportunity.

A new player appeared on the market in 2021 with the launch of grommunio [1], which advertises itself as a largely complete replacement for Microsoft Exchange. Moreover, the startup promises features such as support for the Messaging Application Programming Interface (MAPI)-over-HTTP protocol, file synchronization (ownCloud), chat (Mattermost), and video conferencing with its own Jitsi fork (Figure 1).

Exchange Exasperation

Administrators with mixed feelings about Exchange are reticent to use it, and Microsoft is feeling growing resentment among some users. Reasons include vendor lock-in, a general lack of data protection

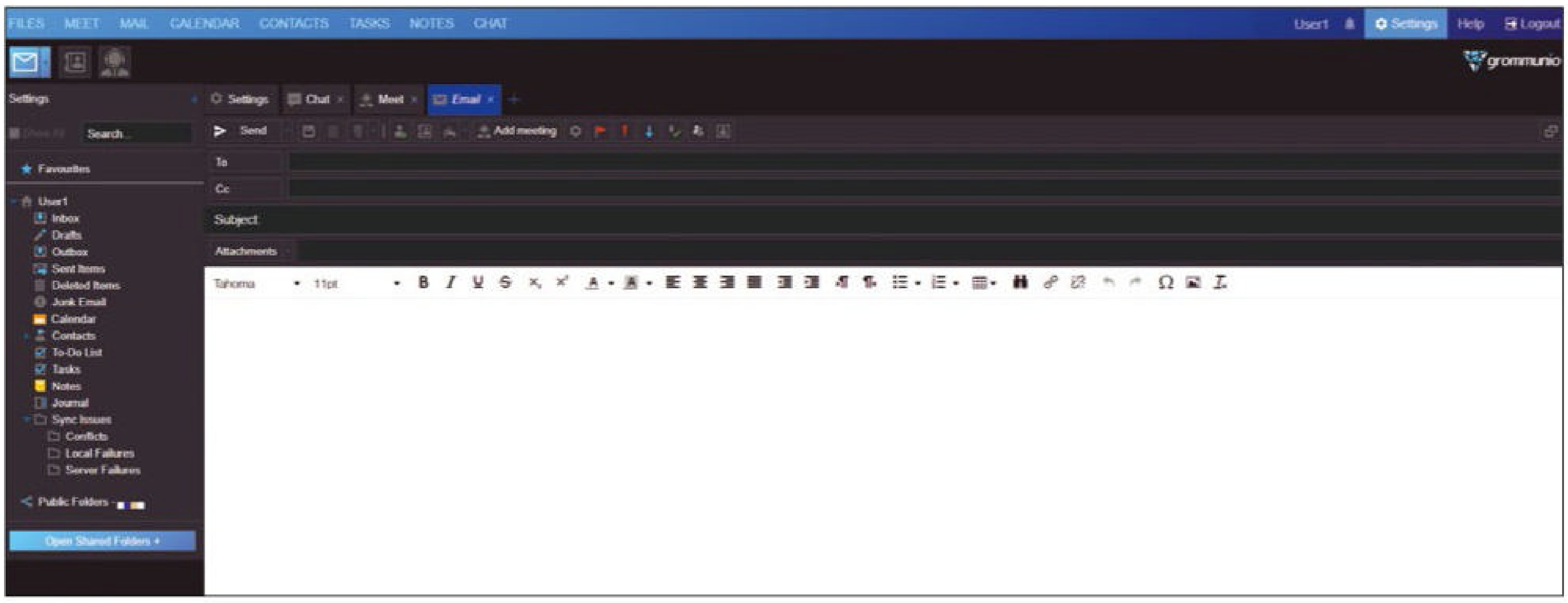


Figure 1: Video, chat, notes, tasks: Grommunio's original name derives from the claim "groupware and much, much more." © grommunio.com

Photo by Gabriel Crismariu on Unsplash

compliance, and the vulnerability of Microsoft's products to malware. Recently, Microsoft had to fix another vulnerability in Exchange that exposed client user data [2]. Many admins feel left out by Microsoft: Policies, strategies, and responses to customer complaints are typically geared for very large installations. For fans of open source, Exchange adds another problem: It is not based on open standards, and it uses proprietary protocols and APIs like MAPI [3] for internal communication. Clients for Exchange are preinstalled on all Windows, Android, and Apple devices, but Exchange still has a poor reputation when it comes to communicating with external services or third-party clients. Microsoft promises to emulate free standards like the Simple Mail Transfer Protocol (SMTP) or Internet Message Access Protocol (IMAP) but lacks the functions necessary for compatibility.

In light of the advance of cloud-based solutions, many companies are under pressure to hand over their data to corporations like Google, Apple, or Microsoft. Microsoft's flagship products are also heading in the direction of cloud-only offerings. Exchange and other products such as MS Office are increasingly being drawn into this vortex. Although Exchange is still available on-premises, Microsoft is increasingly focusing on cloud products [4].

European companies cannot transfer their data to a third-party cloud, even more so if it is hosted in another jurisdiction or by a company based there. The General Data Protection Regulation (GDPR) and liability and data protection considerations prohibit doing so. When it comes to public administration data or particularly sensitive health data,

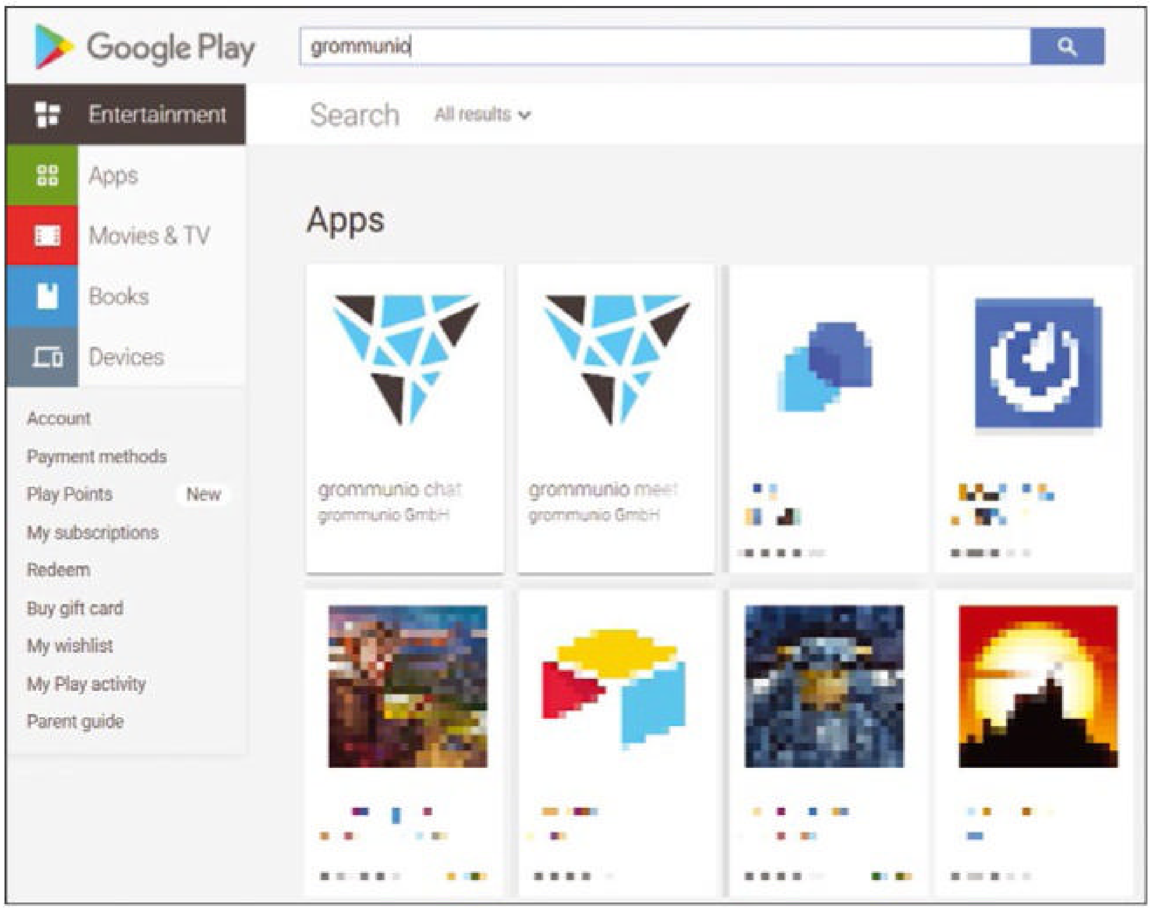


Figure 2: Grommunio is constantly and rapidly evolving: The August 2021 version introduced many new features, including a migration wizard and two apps in the Google Play store for meetings and chats.

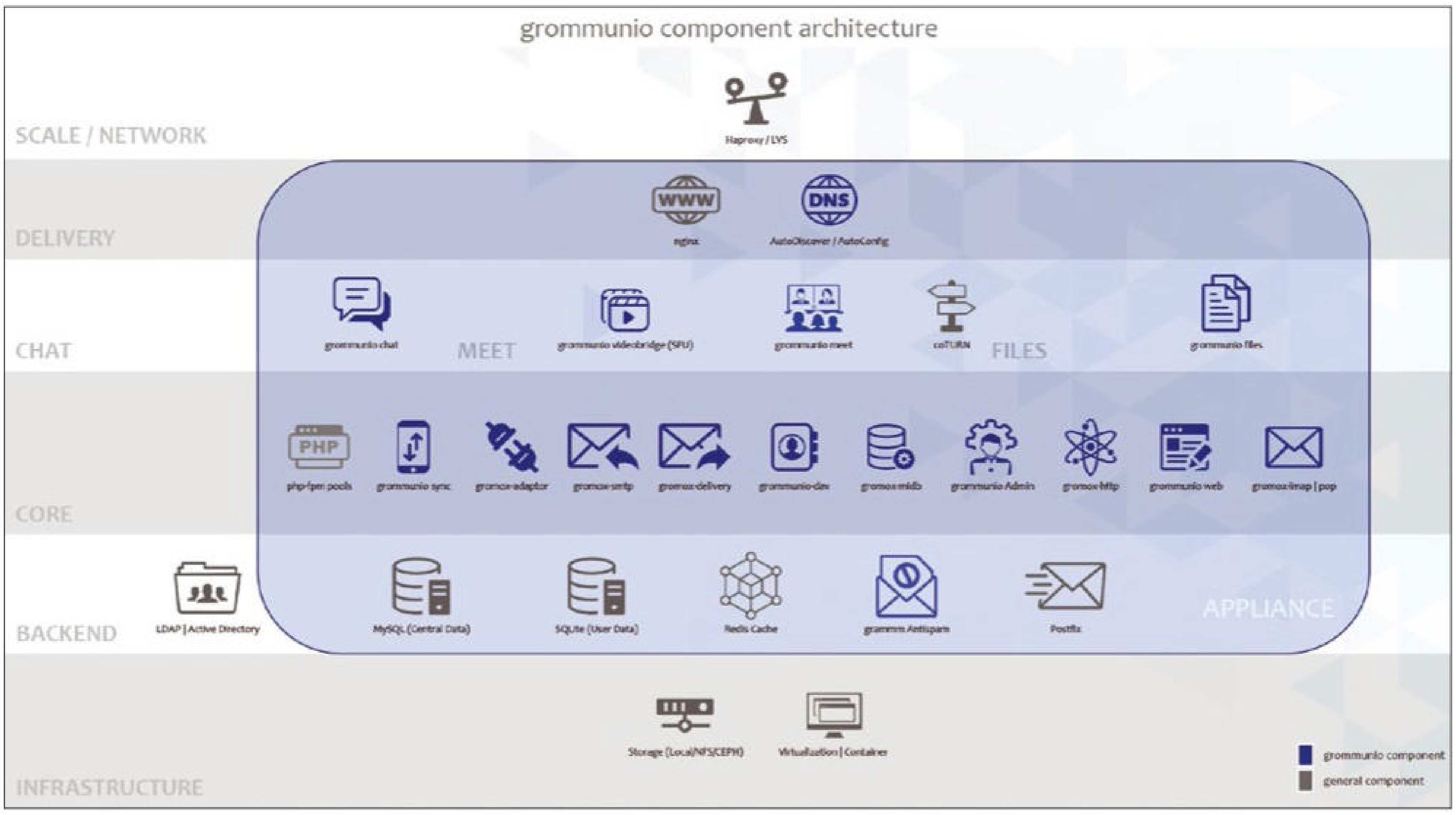
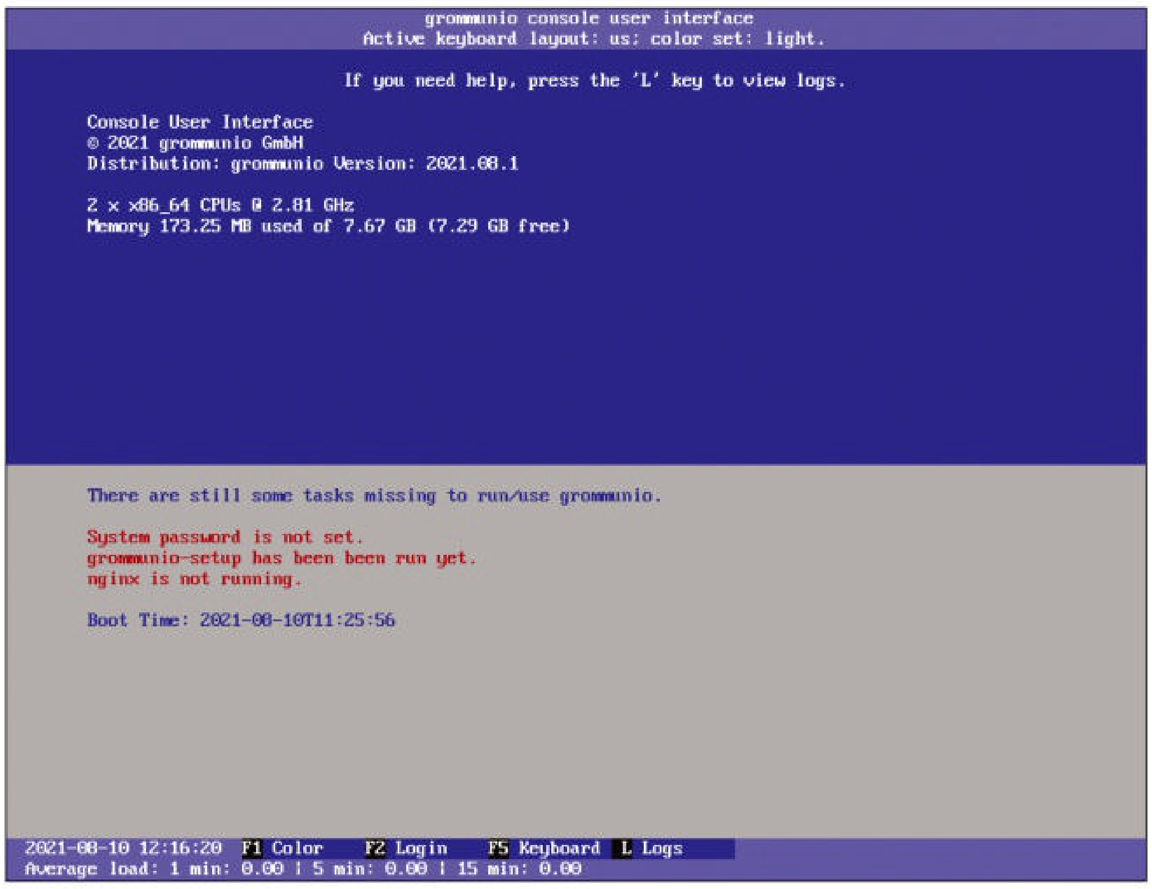


Figure 3: Grommunio's service architecture is based on sharding.

for example, the restrictions are even tighter, leading to a growing market that needs an alternative to Microsoft Exchange, wherein mail is hosted locally and transparently on the company's own servers and is demonstrably under the control of the operator.

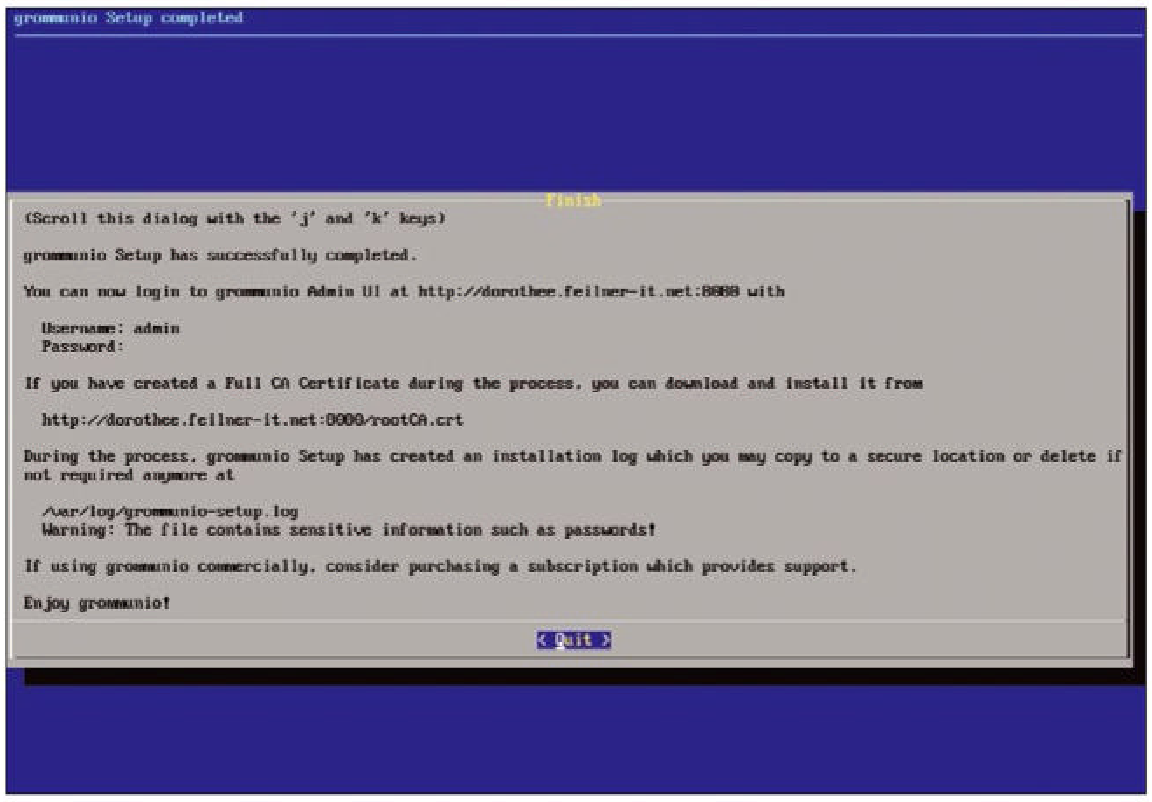
Transparent for Clients

According to the developers, grommunio supports both the aged Remote Procedure Call (RPC)-over-HTTP protocol by Microsoft from 2017 and the newer MAPI over HTTP [5] and offers ActiveSync for mobile devices. In contrast to Exchange, the open source groupware additionally entices users with standard protocols for other clients. The choice extends from CalDAV (calendar) and CardDAV (address book) to a variety of Microsoft Exchange protocol extensions. According to the manufacturer, a grommunio instance can be integrated easily into an existing Windows Active Directory (AD) domain.



```
                    grommunio console user interface
               Active keyboard layout: us; color set: light.

               If you need help, press the 'L' key to view logs.

    Console User Interface
    © 2021 grommunio GmbH
    Distribution: grommunio Version: 2021.08.1

    2 x x86_64 CPUs @ 2.81 GHz
    Memory 173.25 MB used of 7.67 GB (7.29 GB free)

    There are still some tasks missing to run/use grommunio.

    System password is not set.
    grommunio-setup has been been run yet.
    nginx is not running.

    Boot Time: 2021-08-10T11:25:56

 2021-08-10 12:16:20  F1 Color   F2 Login   F5 Keyboard  L Logs
 Average load: 1 min: 0.00 | 5 min: 0.00 | 15 min: 0.00
```

Figure 4: **Before the installation, you need to log in by pressing F2 and possibly selecting the keyboard language with F5.**



```
grommunio Setup completed

                                Finish
(Scroll this dialog with the 'j' and 'k' keys)

grommunio Setup has successfully completed.

You can now login to grommunio Admin UI at http://dorothee.feilner-it.net:8080 with

  Username: admin
  Password:

If you have created a Full CA Certificate during the process, you can download and install it from

  http://dorothee.feilner-it.net:8080/rootCA.crt

During the process, grommunio Setup has created an installation log which you may copy to a secure location or delete if
not required anymore at

  /var/log/grommunio-setup.log
  Warning: The file contains sensitive information such as passwords!

If using grommunio commercially, consider purchasing a subscription which provides support.

Enjoy grommunio!

                               < Quit >
```

Figure 5: **Before the end of the setup, grommunio informs you of your choice of settings.**

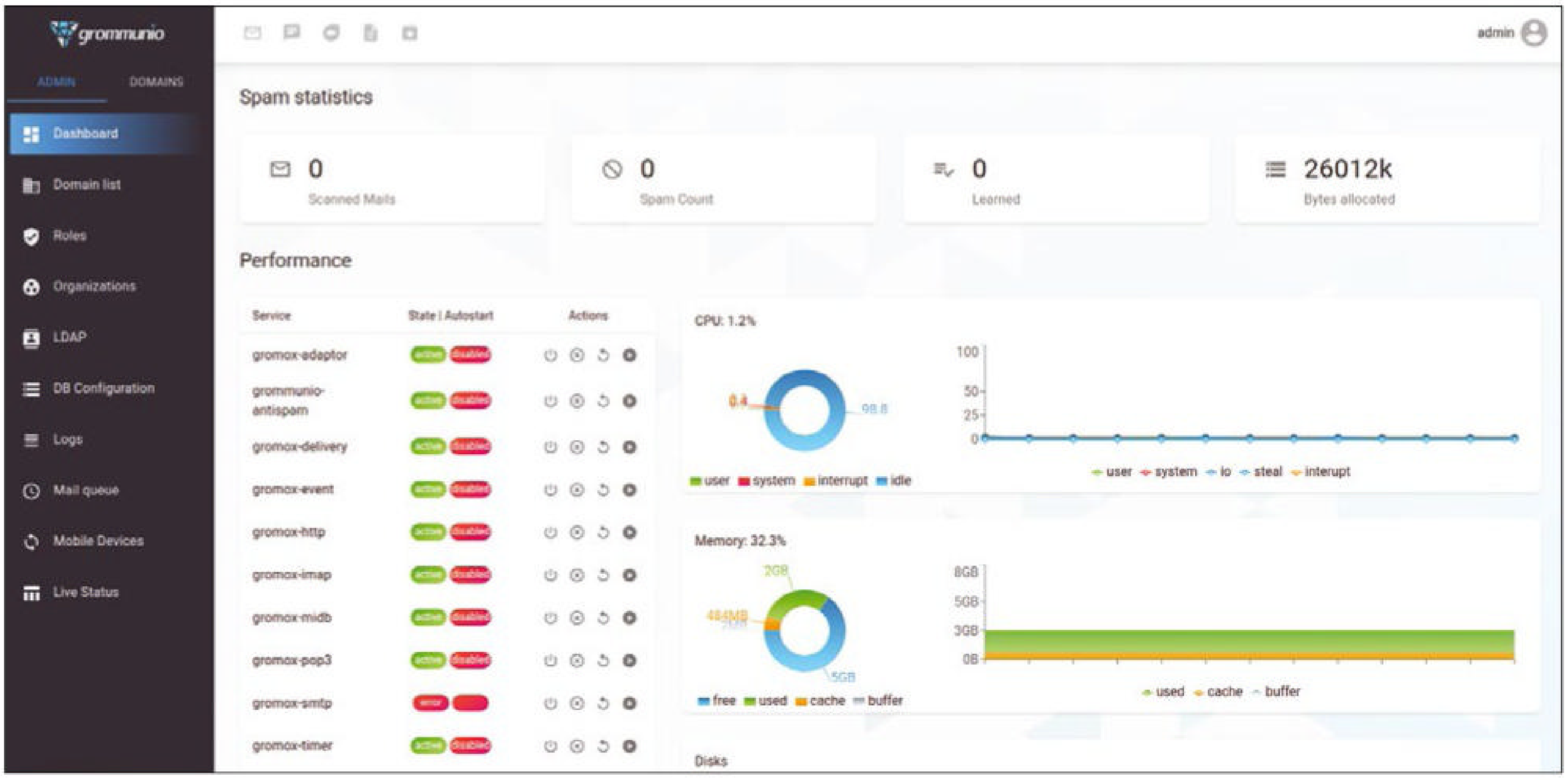
Linux provides the operating system (openSUSE, Ubuntu, Red Hat, Raspberry Pi OS), which makes the solution significantly more robust against malware and hackers. Like its competitors, grommunio comes with a number of typical add-on services that allow you to manage individual notes and tasks; to communicate by chat, video, and audio telephony; and to exchange files with a sync tool. For example, the modular component architecture lets you integrate chat with a simple command like:

```
zypper in grommunio-chat
```

Version 2021.08.1 introduced a menu for the setup routine, from which admins can select or deselect modules. Grommunio claims that the choice of clients for the basic feature set is almost unrestricted – thanks to open standards. However, for the time being, features beyond those offered by Exchange are only available with the web interface. Apps for mobile devices are on the roadmap, and apps already exist for Chat and Meet (Jitsi) in the Google Play store (Figure 2). As a client, grommunio accepts any (mobile) Exchange client that connects by ActiveSync or other protocols. Users do not notice whether they are adding the new account on a real Exchange or on grommunio. After entering a username and password, the client goes about providing service in the usual way. Grommunio plans to provide features not supported by the Microsoft API or vendor protocols – file synchronization, calls, chat, video conferencing – with separate apps.

Scaling by Sharding

Like Exchange, grommunio uses sharding to scale seamlessly (Figure 3). Essentially, this involves distributing horizontally partitioned databases across multiple hosts. Transparent to the user, grommunio can use multiple shards located on different servers by simply adding another server or cloud account to



Service	State | Autostart	Actions
gromox-adaptor	active disabled	
grommunio-antispam	active disabled	
gromox-delivery	active disabled	
gromox-event	active disabled	
gromox-http	active disabled	
gromox-imap	active disabled	
gromox-midb	active disabled	
gromox-pop3	active disabled	
gromox-smtp	error	
gromox-timer	active disabled	

Figure 6: The grommunio server web interface offers extensive monitoring functions.

the storage back end when space becomes scarce.

Unlike its competitors, grommunio uses a MySQL database for the metadata only and stores content data (e.g., mail and groupware objects) in a SQLite database for each user. The platform allows multidomain and multitenant setups and runs with any POSIX-compliant storage back end, according to the vendor.

Getting Started

The software is available on GitHub [6] or as a packaged distribution from grommunio's customer repository service [7] or a community repository service [8]. Internally, grommunio uses the Open Build Service [9].

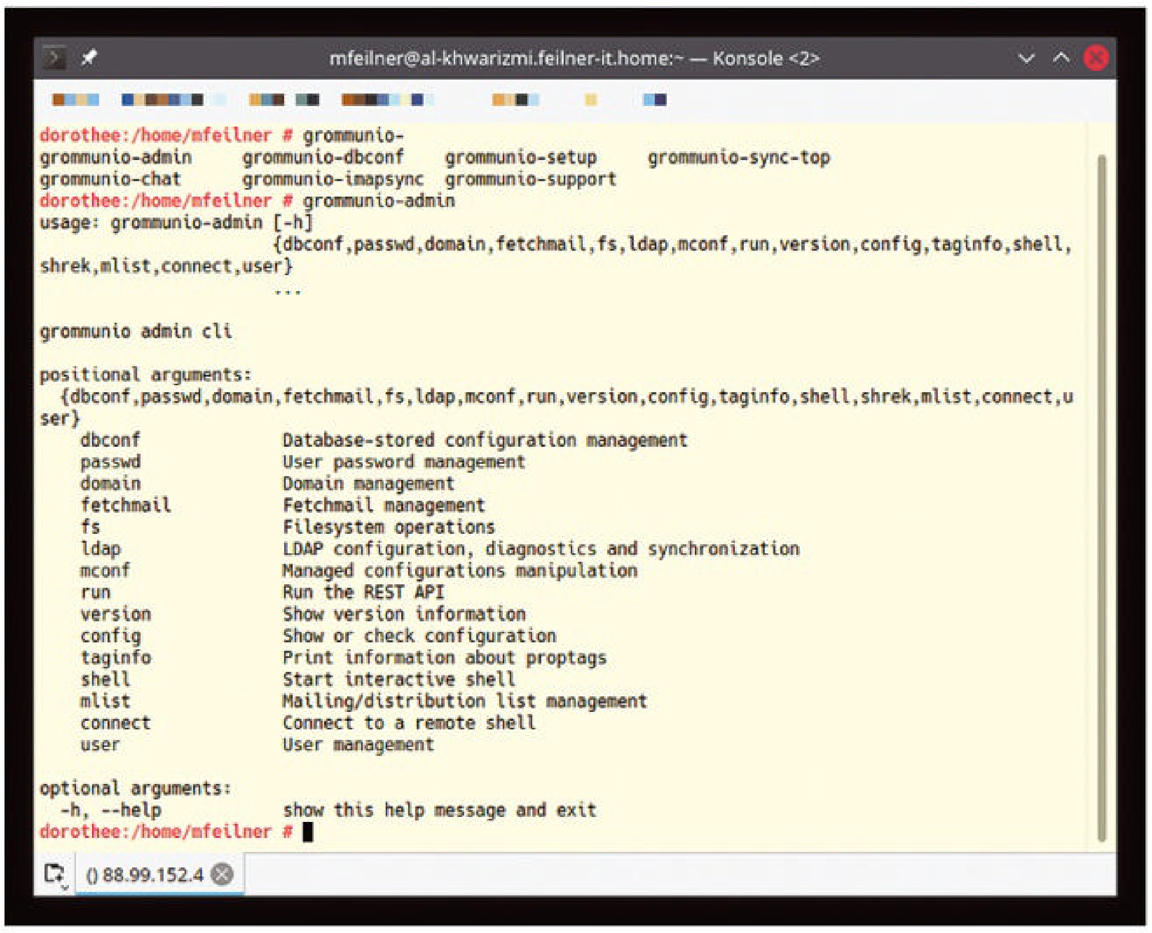
Initially, grommunio provides an ISO file that boots directly into the installation routine. After a reboot, the setup wizard waits for the initial configuration of the new grommunio instance. To begin, you import an existing SSL certificate or create a Let's Encrypt certificate with a supplied script.

Installation, configuration, and debugging take place on the grommunio console interface (Figure 4). The installation wizard asks whether it can delete the data on the first disk it finds. Afterward, grommunio shows you the admin login prompt and the decisions you made (Figure 5).

In the next step, you define further installation parameters like the correct mail domain(s). Regardless of the platform on which grommunio is installed, you can work at the grommunio command-line interface, in the admin graphical user interface (GUI), or from the web interface.

Many options are available both in the browser (Figure 6) and in the shell (Figure 7), and users can enable individual modules for themselves in the web interface (Figure 8). For example, to add users to the mail domain, go to the *Domain list* section in the web interface.

Because the ISO image is built on openSUSE, it uses the distribution's standard tools, including YaST for network configuration and Zypper for package management. The



```
dorothee:/home/mfeilner # grommunio-
grommunio-admin     grommunio-dbconf     grommunio-setup     grommunio-sync-top
grommunio-chat      grommunio-imapsync   grommunio-support
dorothee:/home/mfeilner # grommunio-admin
usage: grommunio-admin [-h]
                       {dbconf,passwd,domain,fetchmail,fs,ldap,mconf,run,version,config,taginfo,shell,
shrek,mlist,connect,user}
                       ...

grommunio admin cli

positional arguments:
  {dbconf,passwd,domain,fetchmail,fs,ldap,mconf,run,version,config,taginfo,shell,shrek,mlist,connect,u
ser}
    dbconf              Database-stored configuration management
    passwd              User password management
    domain              Domain management
    fetchmail           Fetchmail management
    fs                  Filesystem operations
    ldap                LDAP configuration, diagnostics and synchronization
    mconf               Managed configurations manipulation
    run                 Run the REST API
    version             Show version information
    config              Show or check configuration
    taginfo             Print information about proptags
    shell               Start interactive shell
    mlist               Mailing/distribution list management
    connect             Connect to a remote shell
    user                User management

optional arguments:
  -h, --help            show this help message and exit
dorothee:/home/mfeilner # █
```

Figure 7: You can configure numerous settings and install modules at the Linux command line.

```
zypper ref
zypper up
gromox-dbop -U
```

commands update the RPM packages and the database schema. A reboot is recommended afterward.

Performance

The closer the user to the software, the more important the performance for the acceptance of the chosen tool. Numerous failed open source groupware systems have already been forced to learn that a positive user experience with services such as email and calendars is massively dependent on responsiveness and speed. If it takes too long to receive mail, dissatisfaction spreads quickly. Components that cause a delay are discarded because of architectural decisions that imply massive hardware requirements for large setups. Well-coded software is always cheaper than new hardware. Grommunio is aware of this phenomenon: The Vienna-based company refers to up to 50,000 active mailboxes on a high-end server, according to load tests and customer experience. Functional tests currently deliver promising results, which according to the developers is also due to the elegance of MAPI over HTTP: Microsoft has done a good job here, and grommunio also benefits. MAPI over HTTP is load-balanced and proxy-enabled and has proven to be very practical in terms of scalability – a task at which many open source groupware projects have already failed.

Closing the Gaps

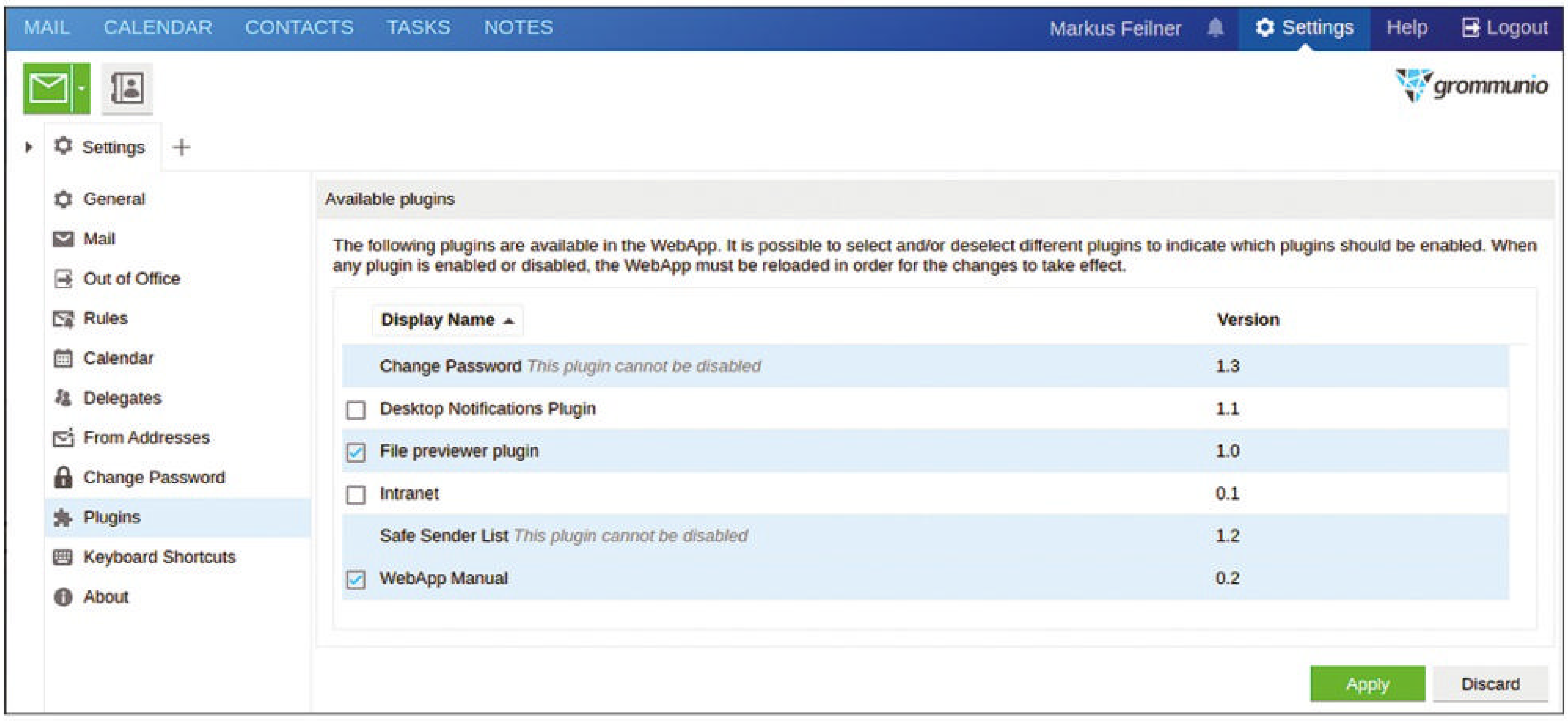
Mail and groupware servers in the Microsoft cosmos are often a cyber battlefield. In the past, the largest malware attacks have always been routed through this infrastructure, and complete protection seems impossible. Although Outlook and Office 365 remain attack vectors with grommunio, two of the main entry points for hackers can be secured in a far better way: the groupware server itself and its operating system. No more Windows, no more waiting for patches and, on top of that, an open source community to help fix bugs and vulnerabilities – that sounds promising.

By default, grommunio runs on many Linux distributions on platforms ranging from the Z-series mainframe to the Raspberry Pi. Given a connection, an Active Directory or Lightweight Directory Access Protocol (LDAP) directories can act as an automatic source for usernames during authentication. Because the server integrates the Pluggable Authentication Modules (PAMs) built into Linux, tokens and all other known PAM solutions also work, provided the client supports challenge-response. The same applies to clients such as Outlook, Thunderbird, Apple Mail, Gmail, or Evolution: Whatever the operating systems allow, authentication works with the grommunio server in the same way as Microsoft Exchange.

Mobile Use

Grommunio's Mobile Device Management (MDM) is based on the features included in Microsoft Exchange. Together with grommunio Sync, it takes care of sharing and synchronizing mailboxes and enables access to shared folders depending on the user's authorization. Additionally, remote wipe is supported, with device policies per the manufacturer. Remote wipe and some other functions can be configured in the administrator panel in the web GUI without user interaction.

MDM does not have a separate app because, just like Exchange, grommunio uses the functions of Microsoft Exchange clients, which are disguised as "accounts" through provisioning in mobile devices. After setting up an account, the administrator can remotely wipe the phone. Users can also delete and reconnect,



Available plugins

The following plugins are available in the WebApp. It is possible to select and/or deselect different plugins to indicate which plugins should be enabled. When any plugin is enabled or disabled, the WebApp must be reloaded in order for the changes to take effect.

	Display Name ▲	Version
	Change Password *This plugin cannot be disabled*	1.3
□	Desktop Notifications Plugin	1.1
☑	File previewer plugin	1.0
□	Intranet	0.1
	Safe Sender List *This plugin cannot be disabled*	1.2
☑	WebApp Manual	0.2

Figure 8: Whether desktop notifications or file previews, users can configure their modules in the web interface.

just as when using an Exchange server. The ownCloud-based grommunio Files function stores data on users' devices, enabling sharing, syncing, offline use, and many other features, including a search engine, versioning, and automatic image upload.

Data Migration

A simple definition of digital sovereignty is: "exit strategy first." Before signing a contract, IT management must answer the question of how to migrate data away from a new server product. Grommunio seems promising here with its open standards. For example, email can be read with any standards-compatible mail client. Because the groupware product is open source, server states can be tracked, data model changes can be analyzed, and actions can be initiated.

Things don't always work that seamlessly with Microsoft Exchange. Exchange admins know that, in the past, even migrating from one version to the next could be tedious and hazardous, sometimes even failing completely. Although Microsoft upgrades have certainly improved in quality and stability, migrations to third-party groupware solutions at the latest almost always rely on the physical presence of the admin in front of the client.

Scripts offered a remedy to this problem (e.g., for Microsoft PowerShell, which the grommunio developers also use as a basis). The `gromox-pffimport` tool has been added to their product and offers scripts for importing Exchange data by user and account. Since 2022, users benefit from the powerful PowerShell `exchange2grommunio.ps1` script [10] that connects to the server and handles the migration. The script automatically creates mailboxes on the destination server and uses the `plink.exe` executable for secure connections, Common Internet File System (CIFS) for access, and `.pst` files for data migration. Administrators will have to fill in some variables in the script and then run the script from the Exchange admin shell. The script has about 500 lines, assumes a correct LDAP/AD setup (also with grommunio attached!), and requires at least Windows Server 2012 R2 and PowerShell 2.0. Listing 1 shows the first lines of the migration loop.

Migration Tools

So far, an alternative way to migrate data has been lacking. Like the majority of groupware products on the market, Microsoft Exchange uses its own formats to store user data such as mailboxes, calendar entries, and

Listing 1: Migration by Script

```
foreach ($Mailbox in (Get-Mailbox)) {
  $MigMBox = $Mailbox.PrimarySmtpAddress.ToString()
  if ($IgnoreMboxes.contains($MigMBox)) {
    $MailboxesSkipped++
    $MailboxesTotal++
    Write-Host "Ignoring mailbox: $MigMBox" -fore yellow
    continue
  }
  Write-Host ""

  [...]

  Write-Host "Removing all MailboxExportRequests." -fore green
  Get-MailboxExportRequest | Remove-MailboxExportRequest -Confirm:$false

  # Remove old / orphaned .pst file
  if (Test-Path -Path $WinSharedFolder\$MigMBox.pst) {
    Remove-Item -ErrorAction SilentlyContinue -Path $WinSharedFolder\$MigMBox.pst
    Write-Host "Removing outdated $MigMBox.pst file." -fore yellow
  }
  Write-Host ""

  # Create a .pst file for every mailbox found on system.
  #
  Write-Host "Exporting mailbox $MigMBox to file $MigMBox.pst..." -fore green

  [...]

  New-MailboxExportRequest -Mailbox $Mailbox -FilePath $WinSharedFolder\$MigMBox.pst | ft
    -HideTableHeaders
  Write-Host -NoNewline "[Wait] " -fore yellow
  $MailboxesTotal++

  [...]

  $nTimeout = 0
  while ((Get-MailboxExportRequest -Mailbox $Mailbox).Status -ne "Completed") {
    Start-Sleep -s 2
    $nTimeout += 2
    if ($nTimeout % 60 -eq 0) {
      Write-Host -NoNewline "|" -fore yellow
    } else {
      if ($nTimeout % 10 -eq 0) {
        Write-Host -NoNewline "." -fore yellow
      }
    }
  }

  [...]
```

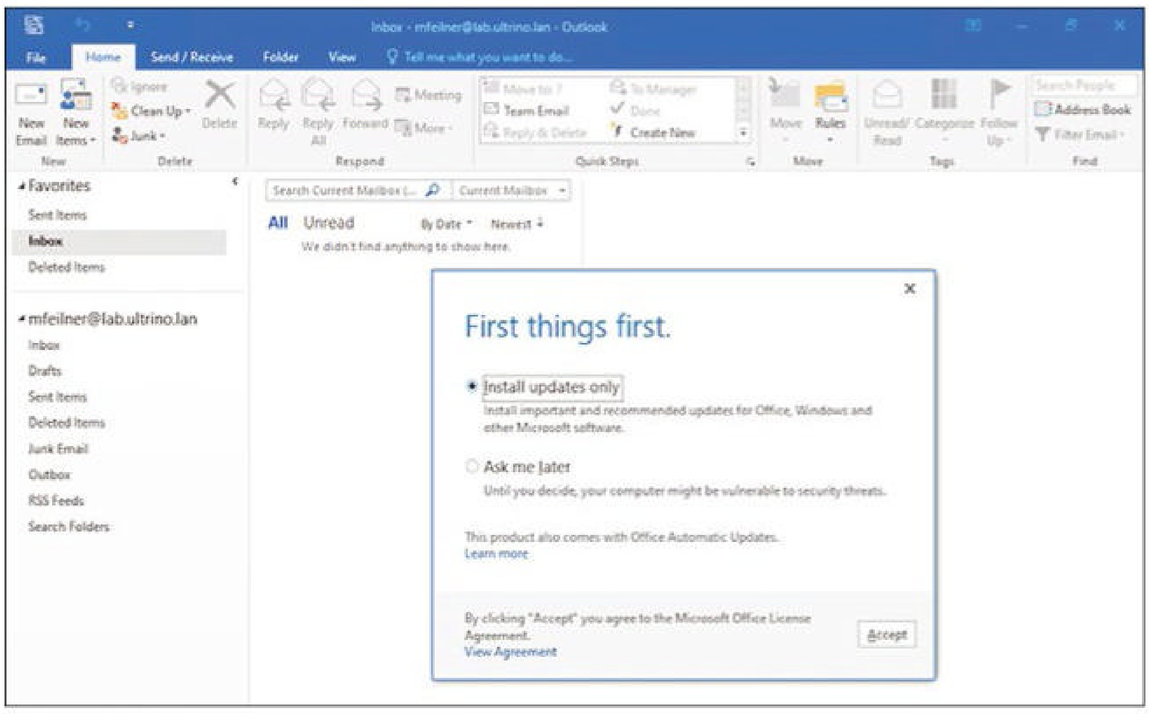
contacts. To exchange information between different instances in a better way, it does let you export files to compatible formats. Grommunio uses a similar approach to import user data exported from Exchange into its user database Gromox.

The grommunio developers are aware of the importance of a migration tool, especially given their ambitious claim of being able to act as a drop-in replacement for Exchange. Grommunio supports the Microsoft PST (Outlook data file), OST (offline Outlook data file), and PAB (personal address book) formats and integrates the open source `imapsync` IMAP transfer tool and `fetchmail`. Both are used to import data from other servers. A server-side migration tool is already under development.

What the Future Holds

Only a few months old, grommunio is already quite far down the road to becoming a true drop-in replacement for Exchange. Unlike its predecessors, it bundles open source and open standards services and promises to lead the way out of the vendor lock-in jungle. So far, however, admins are likely to be missing more detailed documentation [11] and a full-fledged, automated migration wizard. Grommunio is aware of these shortcomings and is already working intensively on both topics, as the latest updates on the website confirm. Moreover, the migration tools are currently updated frequently.

The Outlook connection (Figure 9) works without problems (Figure 10) but will still have to prove its capabilities in the coming months. The comparatively low prices (see the “Editions” box and Table 1) and the open source character of what is still fairly new software are likely to lower the inhibition threshold for users wanting to try out the groupware product. Extensive independent tests are still outstanding and would be important to back up the vendor's claims relating to speed, stability, scalability, and migration options. If grommunio stands up to the test and the developers deliver the missing features, then grommunio could become a real game changer on the groupware market. The approach definitely looks promising. ■



We didn't find anything to show here.

First things first.

● Install updates only

Install important and recommended updates for Office, Windows and other Microsoft software.

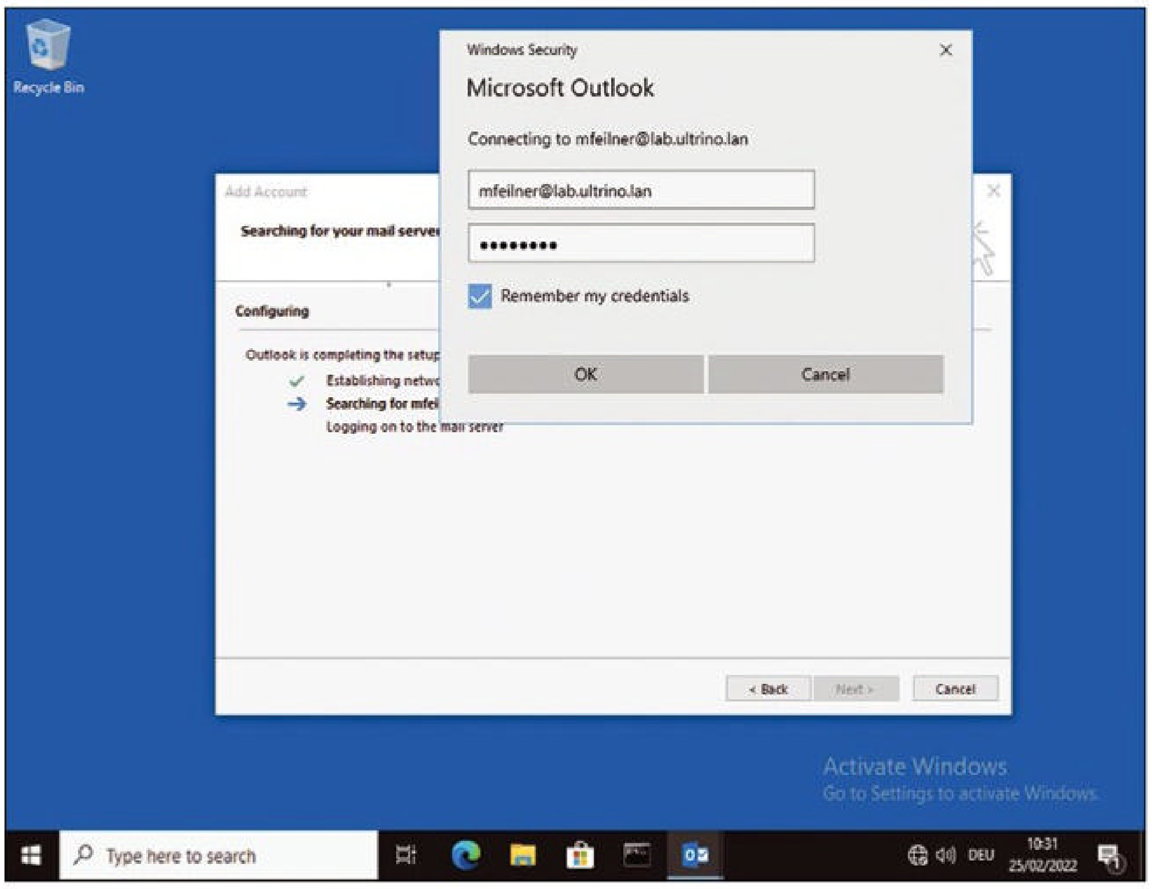
○ Ask me later

Until you decide, your computer might be vulnerable to security threats.

This product also comes with Office Automatic Updates.
Learn more

By clicking "Accept" you agree to the Microsoft Office License Agreement.
View Agreement

Figure 9: **After creating users and domains, connecting Outlook users is a smooth process. Microsoft's email client thinks it is talking to an Exchange server.**



Searching for your mail serve

Configuring

Outlook is completing the setup

✓ Establishing netw

→ Searching for mfei

Logging on to the mail server

Windows Security

Microsoft Outlook

Connecting to mfeilner@lab.ultrino.lan

mfeilner@lab.ultrino.lan

••••••••

☑ Remember my credentials

Figure 10: **Outlook users are connected natively in grommunio without a client-side plugin.**

Info

[1] grommunio: [http://grommunio.com/]

[2] Autodiscover flaw in Microsoft Exchange leaking credentials: [https://www.techtarget.com/searchsecurity/news/252507119/Autodiscover-flaw-in-Microsoft-Exchange-leaking-credentials]

[3] MAPI: [https://en.wikipedia.org/wiki/MAPI]

[4] End of REST API on-premises mailboxes preview: [https://techcommunity.microsoft.com/t5/exchange-team-blog/the-end-of-the-rest-api-for-on-premises-mailboxes-preview/ba-p/3221219]

[5] MAPI over HTTP: [https://docs.microsoft.com/en-us/Exchange/clients/mapi-over-http/mapi-over-http?view=exchserver-2019]

[6] GitHub: [https://github.com/grommunio]

[7] Repository: [https://grommunio.com/download/]

[8] Community repositories: [https://download.grommunio.com/community/packages/]

[9] grommunio openSUSE Build Server: [https://build.grommunio.com]

[10] PowerShell migration script: [https://github.com/grommunio/gromox/blob/master/tools/exchange2grommunio.ps1]

[11] Documentation: [https://docs.grommunio.com]

Author

Markus Feilner, technology and network policy editor at Mailbox.org, has been working with Linux since 1994. He was deputy editor-in-chief of Linux Magazine and iX and doc team leader at SUSE. He has run his company Feilner IT, specializing in documentation, digital sovereignty, and OSI layers 8, 9, and 10, for 22 years.

Editions

Grommunio's free Community subscription provides accounts for five users with basic features such as groupware and mobile device management. The Basic version includes email or web support. Professional users who need telephony support need to purchase the Plus edition. For enterprise customers, the Business subscription has all features, including high availability. Customers who need round-the-clock support have to request this, and it is the only difference between the business and enterprise licenses, apart from the targeted company size. For science, education, public service, government, and NGOs, grommunio grants discounts of up to 25 percent. Hosters can also receive a licensing program specifically tailored to their needs.

Table 1: Features at a Glance

Manufacturer	grommunio GmbH
License	AGPLv3
Price (per user/month)	Community (5 users or fewer) free, Basic (6-49) EUR1.99, Plus (6-49) EUR2.99, Business (50-999) EUR4.49, Enterprise (1000 or more) ask
Operating models	On-premises, cloud
Server OS	Linux
Groupware protocols	SMTP, IMAP, POP3, MAPI, RPC over HTTP, CalDAV, CardDAV, WebDAV, LDAP, Active Directory, Exchange ActiveSync (EAS), Exchange Cached Mode
Exchange Features	
Email	Offline mode, public folders, categories and flags, webmail, global address lists
Contacts	Support for all contact fields, multiple contact folders, contact groups, export
Calendar	Free/busy times, appointment invitations, recurring appointments and exceptions, shared calendars and delegations
Groupware Features	
Features	Notes, tasks (with time tracking), file synchronization and sharing (ownCloud), chat (Mattermost), video conferencing (Jitsi), office features (OnlyOffice), archive
Clients	Outlook (without plugin), all Exchange clients, web GUI, app (chat, video, files), email clients, calendar clients, global search
Other	
Architectures	High availability, sharding, multidomain, multitenancy, all POSIX-enabled storage back ends, AD forest installations
Security	S/MIME, remote wipe, OpenPGP, single sign-on (in development), two-factor authentication (2FA; in development), anti-spam, antivirus (third party)
Administration Features	
Installers	Installer ISO, OVA, image
MDM	Yes
Data migration	Export from Exchange/Kopano/ICAL/IMAP/POP3 to grommunio
White label/branding	Yes
App stores	No

Setting up a PXE boot server

Remote Starter

Set up a PXE server for BIOS and UEFI clients and use it to boot Linux or Windows.

By Andreas Stolzenberger

The preboot execution environment (PXE) lets you boot computers and virtual machines over a network. In this article, I describe how to set up a PXE server, go into detail about the role of the Dnsmasq service, and describe how to make individual PXE configurations.

BIOS

When a computer, whether physical or virtual, is switched on, a number of programs are executed before the operating system even starts. What the mainframe world refers to as initial program load (IPL) is commonly known in the PC world as booting. Technically speaking, this means that the processor sets the program counter to memory cell 0 and then works its way forward in memory until it finds executable program code. On a regular PC, what the CPU encounters first is the Basic Input Output System (BIOS) or, to be more precise, multiple BIOSs, because every device that is plugged into a PC is allowed to mount its own BIOS in memory. This code then runs before the operating system and initializes the hardware. In a desktop PC, the graphics card BIOS is extremely important; otherwise, the screen would stay blank. On servers, which can get by without a graphics card if need be, a redundant array of independent disks (RAID) or storage area network (SAN) controller might need to run its BIOS first so that the booting operating system can find a hard disk. As early as in the 1980s, an option was introduced to run the system boot over a local area network (LAN) with a BIOS on the network interface card (NIC). At that time, however, competing NIC manufacturers used proprietary boot code. It wasn't until 1998 that Intel introduced the PXE 2.0 specification, which has been used for all systems ever since.

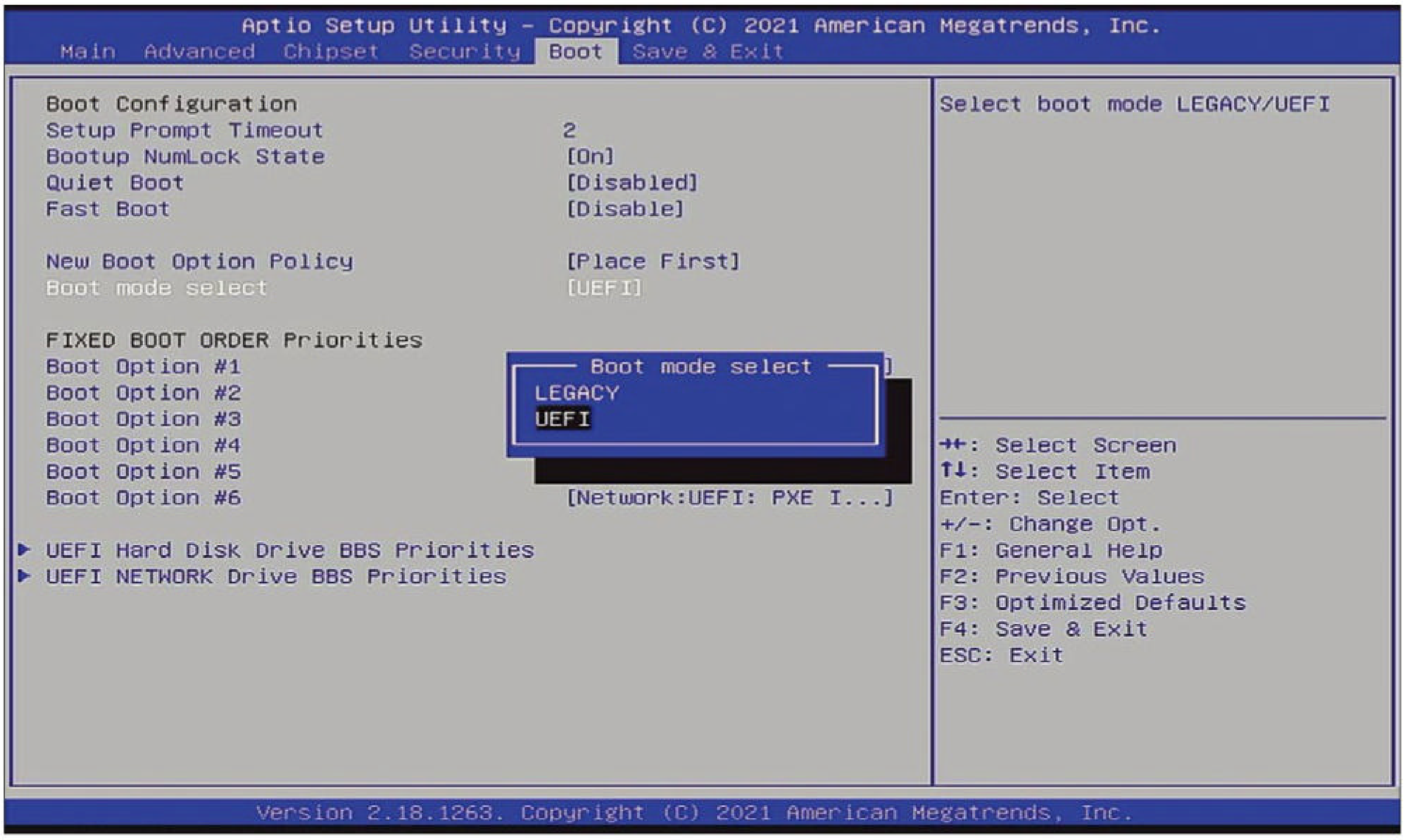
PXE has evolved in the meantime. Strictly speaking, two different network boot processes are in use: PXE for PCs with BIOS and PXE for systems with (Unified) Extensible Firmware Interface (U)EFI (Figures 1 and 2). In this article, I go into more detail about both processes.

PXE Basics

Every modern PC or server with a built-in network card can now boot its operating system over the LAN with PXE. Administrators primarily use this function to install new computers, but PXE can do more. Users can use PXE to boot diskless workstations or even computers that use an Internet SCSI (iSCSI) SAN drive as a disk, which naturally gives you an excellent disaster recovery option. Users can simply push continuous backups of their local drive onto an iSCSI volume. If the local disk fails, they simply boot the system by PXE and use the iSCSI copy as the root drive.

The PXE method is based on a number of standard IP protocols, starting with the Dynamic Host Configuration

Photo by Kiran CK on Unsplash



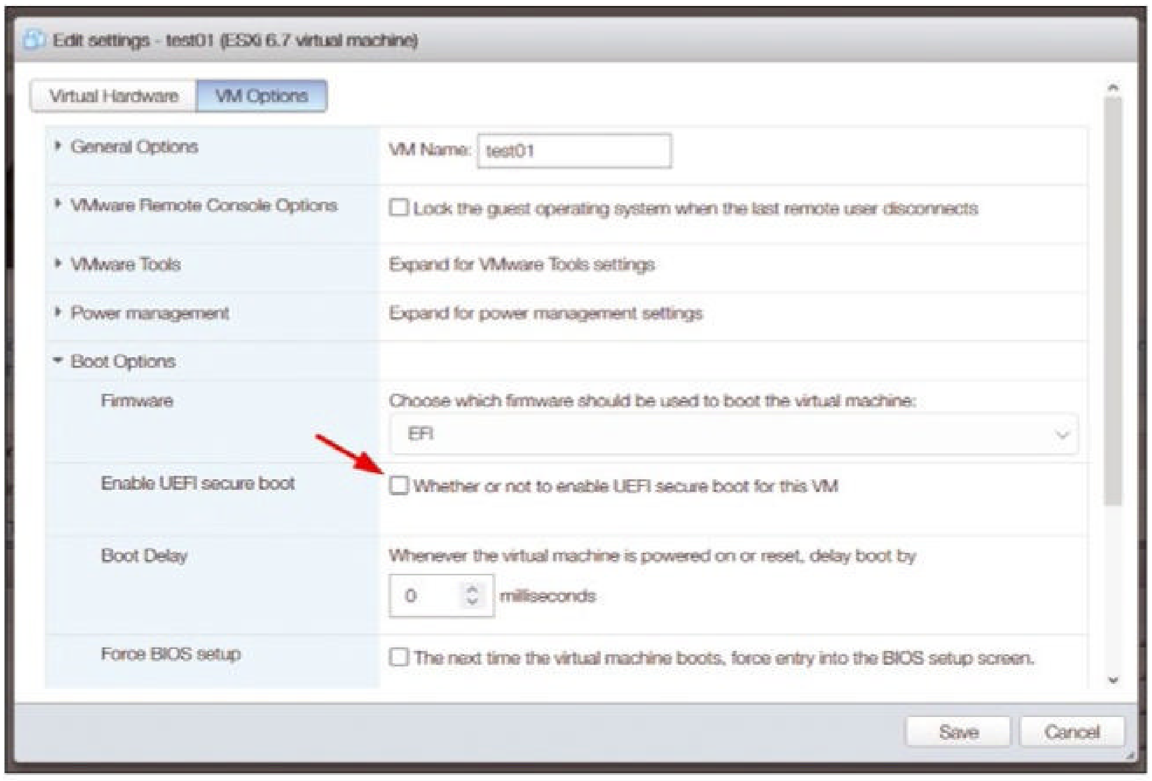
```
        Aptio Setup Utility – Copyright (C) 2021 American Megatrends, Inc.
   Main  Advanced  Chipset  Security  Boot  Save & Exit

 Boot Configuration                                  Select boot mode LEGACY/UEFI
 Setup Prompt Timeout             2
 Bootup NumLock State             [On]
 Quiet Boot                       [Disabled]
 Fast Boot                        [Disable]

 New Boot Option Policy           [Place First]
 Boot mode select                 [UEFI]

 FIXED BOOT ORDER Priorities
 Boot Option #1                ┌── Boot mode select ──┐]
 Boot Option #2                │ LEGACY               │
 Boot Option #3                │ UEFI                 │
 Boot Option #4                └──────────────────────┘  →←: Select Screen
 Boot Option #5                                          ↑↓: Select Item
 Boot Option #6                   [Network:UEFI: PXE I...]  Enter: Select
                                                         +/-: Change Opt.
▶ UEFI Hard Disk Drive BBS Priorities                    F1: General Help
▶ UEFI NETWORK Drive BBS Priorities                      F2: Previous Values
                                                         F3: Optimized Defaults
                                                         F4: Save & Exit
                                                         ESC: Exit

      Version 2.18.1263. Copyright (C) 2021 American Megatrends, Inc.
```

Figure 1: **Most PC BIOSs let you choose between UEFI and BIOS (legacy) boot for the system start, which changes the way the PC then boots via PXE.**

Protocol (DHCP). The client first sends a DHCPDISCOVER message as a Layer 2 broadcast – because it does not yet have an IP address – and waits for the response from a DHCP server. In this first request, the client also transmits its system type (i.e., whether it is booting from BIOS or UEFI). The DHCP server responds to the request with a DHCPOFFER, which contains a good bit of information: the IP address the client is allowed to use and the lease time (i.e., how long it is allowed to keep this address), plus masses of network details relating to the network mask, routes, and DNS servers.

Depending on the services on the LAN, the DHCP server can also provide information about Active Directory (AD), the time servers, or other information. If a boot server for PXE resides on the network, DHCPOFFER sends both the server's IP address and details of the boot code. The PXE boot server can run on a machine other than the DHCP server. In this example, the two services are running together. The client responds to the "offer" with a "request" stating that it wants to keep the offered IP address, and DHCP concludes the process with an "acknowledge."

If the client now wants to boot off the LAN, it first uses the Trivial File Transfer Protocol (TFTP) in both boot methods to fetch the start code from the boot server. TFTP is basically the junior sibling of the well-known File Transfer Protocol (FTP), except that TFTP does not support authentication or any kind of security function (e.g., encryption). Because the protocol uses a very small block size, it limits the transfer size for files to a maximum of 4GB, and it is also quite slow, although this is not a problem, seeing as just a few kilobytes are typically



Edit settings - test01 (ESXi 6.7 virtual machine)

Virtual Hardware | VM Options

- General Options — VM Name: test01
- VMware Remote Console Options — □ Lock the guest operating system when the last remote user disconnects
- VMware Tools — Expand for VMware Tools settings
- Power management — Expand for power management settings
- Boot Options
  - Firmware — Choose which firmware should be used to boot the virtual machine: EFI
  - Enable UEFI secure boot — □ Whether or not to enable UEFI secure boot for this VM
  - Boot Delay — Whenever the virtual machine is powered on or reset, delay boot by 0 milliseconds
  - Force BIOS setup — □ The next time the virtual machine boots, force entry into the BIOS setup screen.

Figure 2: **Hypervisors, such as VMware ESXi shown here, support both UEFI and BIOS boot modes for PXE virtual machine boot.**

Integrating the Domain Controller

If you are running an AD domain controller (DC) on your LAN, you need to create a set of entries in **/etc/dnsmasq.conf** on the DHCP server to enable AD clients to find the DC. This example assumes that the IPv4 domain is named **domain.ip**, the AD domain is named **DOMAIN.IP**, and the domain controller is **adc.domain.ip**.

Of course, you then need a matching entry in **/etc/hosts** to assign the IP address:

```
srv-host =_kerberos._udp.domain.ip, adc.domain.ip,88
srv-host =_kerberos._tcp.domain.ip, adc.domain.ip,88
srv-host =_kerberos-master._tcp.domain.ip, adc.domain.ip,88
srv-host =_kerberos-master._udp.domain.ip, adc.domain.ip,88
srv-host =_kpasswd._tcp.domain.ip, adc.domain.ip,88
srv-host =_kpasswd._udp.domain.ip, adc.domain.ip,88
srv-host =_ldap._tcp.domain.ip, adc.domain.ip,389
txt-record=_kerberos.domain.ip,“DOMAIN.IP“
```

all you need to transfer a kernel or a bootloader. TFTP usually only transfers the first bootloader anyway. Afterward, the procedure can switch to a regular IP protocol such as HTTP, HTTPS, (S)FTP, or even NFS.

Putting the Boot Server into Operation

To leverage the full functionality of PXE and DHCP, you need your own DHCP server that is based on a Linux computer or a Linux virtual machine (VM). Normally, you will already have a DHCP server on the LAN. In smaller networks, the router itself is more likely to handle this role. Of course, you can only have one DHCP server on a LAN. You will need to disable the existing DHCP server, if any, and transfer this task to the new server.

On your system of choice, first install the *dnsmasq* and *syslinux* packages. Next, create a directory for the TFTP server, typically in /var/lib/tftpboot. Then, copy the complete content of the Syslinux installation (in /usr/share/syslinux on Fedora) into the tftpboot directory. You only need this step for PXE with BIOS PCs.

Listing 1: pxelinux.cfg/default Entry

```
label fedora34-live
  menu label Fedora 34 Workstation LiveBoot
  kernel http://<IP address of the DHCP server>/f34/images/pxeboot/vmlinuz
  append initrd=http://<IP address of DHCP server>/f34/images/pxeboot/initrd.img root=live:http://<IP
     address of DHCP server>/f34/LiveOS/squashfs.img ro rd.live.image rd.luks=0 rd.md=0 rd.dm=0
```

Now, install a web server such as Apache or Nginx with the default configuration. In the remainder of this article, I assume that the web server serves up the files from /var/www/html and its subdirectories in the basic configuration.

Configuring Dnsmasq

The Dnsmasq service in the setup I describe here handles all DNS by DHCP and TFTP functions. It can be configured by editing two files: /etc/dnsmasq.conf and /etc/hosts. (See also the “Integrating the Domain Controller” box.) The /etc/hosts file has the names of the local systems for the DNS service, for example:

```
192.168.2.100 server1.domain.ip server1 ads
```

The format starts with the IP address, followed by one or more hostnames, and one entry is the fully qualified domain name (FQDN). Moreover, you need to add an /etc/nameservers.conf file on the server, where you list Internet name servers to which the Dnsmasq service forwards external name requests. You need to enter your provider’s DNS server addresses here – or those of free DNS providers such as Google. The dnsmasq.conf file then looks something like:

```
resolv-file=/etc/nameservers.conf
interface=eno1
dhcp-range=192.168.2.201, 192.168.2.250,72h
```

The interface line must contain the DHCP server NIC. The dhcp-range line specifies an address pool from which the DHCP clients obtain their IP addresses. The lease time of 72 hours here specifies how long DHCP clients can keep their addresses. If your provider has assigned you an IPv6 subnet, your DHCP server can pass on addresses from this segment to your LAN systems, which is sometimes referred to as “router advertisement”:

```
enable-ra
dhcp-range=tag:eno1,::1, ↩
           constructor:eno1, ↩
           ra-names, 12h
```

If needed, you can specify additional DHCP options (according to RFC2132) that you want to pass to your clients, such as a NetBios name server:

```
dhcp-option=44,192.168.2.100
```

Dnsmasq also supports the more readable form:

```
dhcp-option=option:netbios-ns,192.168.2.100
```

The PXE configuration for the BIOS PXE module initially only comprises three lines:

```
enable-tftp
tftp-root=/var/lib/tftpboot
dhcp-boot=pxelinux.0
```

The pxelinux.0 file is the bootloader from the Syslinux package, which is then executed on the starting client. You could use the GRUB bootloader instead. In this article, I use it later for the UEFI boot example.

The PXE server gives the Syslinux bootloader a configuration file. To do this, create a pxelinux.cfg directory in /var/lib/tftpboot. In the directory, create a text file named `default` with the content:

```
MENU TITLE PXE Boot
TIMEOUT 200
TOTALTIMEOUT 6000
ONTIMEOUT local
LABEL local
      MENU LABEL (local)
      MENU DEFAULT
      LOCALBOOT 0
```

The entries are not case sensitive. The standard menu only gives users one option, starting from the system's local disk. A number of options make the PXE menu a bit more colorful and load a PNG image as background, but I will not go into them in detail here. The documentation for the syntax of a Syslinux menu can be found on the Syslinux wiki [1].

Starting Fedora with PXE

To boot a Fedora 34 Live system by PXE, first unpack the content of a Fedora 34 Live ISO image into a subdirectory of your web server:

```
mount -o loop Fedora-Workstation-↩
  Live-x86_64-34-<x.x>.iso /mnt
mkdir /var/www/html/f34
rsync -avx /mnt/ /var/www/html/f34/
```

Next, create the appropriate entry in `/var/lib/tftpboot/pxelinux.cfg/default` (**Listing 1**). After booting a client by PXE, you will see a selection menu for *local* (i.e., starting from the local hard drive or `fedora34-live`). You can boot other Live distributions such as Debian or Ubuntu in a similar way, and you can create kickstart files that automate the installation of a Linux system to enable a fully automatic install from the PXE menu.

The menu does require the use of a modern PXE client that supports the HTTP protocol. Some older PXE implementations cannot handle the `kernel http://...` line. In a test setup, for example, this was the case with the PXE implementation in VirtualBox. To work around this, you need to copy the referenced files `vmlinuz` and `initrd.img` to a subdirectory in `/var/lib/tftpboot` and load them from there over TFTP instead of by HTTP. Assuming you create copies of the files in `/var/lib/tftpboot/f34`, the entry would be:

```
label fedora34-live via TFTP
  menu label Fedora 34 Workstation ↩
    LiveBoot
  kernel f34/vmlinuz
  append initrd=f34/initrd.img ↩
     root=live:http://<IP address of ↩
     DHCP server>/f34/LiveOS/squashfs.img ↩
     ro rd.live.image rd.luks=0 ↩
     rd.md=0 rd.dm=0
```

By the way, the reference to `squashfs.img` by HTTP can be kept because it is evaluated by the booted kernel and not by the PXE loader.

Starting Windows by PXE

To start Windows with a Linux PXE server, you first need a Windows preinstallation environment (PE) ISO image. You will find instructions for setting up the image online [2]. It is important that you add all network drivers that are used on your systems to the PE image. Store the finished Windows PE image in the `/var/lib/tftpboot` directory of your TFTP server. For this example, I'll just dub the image `winpe.iso`. Next, add the following entry to your PXE start menu:

```
label windows
      kernel memdisk
      initrd winpe.iso
      append iso raw
```

Alternatively, extract the content of the Windows PE CD to a subdirectory on the TFTP server (e.g., `/var/lib/tftpboot/pe`) and get the bootloader `wimboot` from the iPXE package [3]. The entry now looks like this:

```
label wimboot
  kernel wimboot
  com32 linux.c32
  append wimboot ↩
    initrdfile=pe/bootmgr,pe/boot/bcd,↩
    pe/boot/boot.sdi,pe/sources/boot.wim
```

The Windows PE instance can now be launched directly over the network, and you can run diagnostic tools or a Windows setup over the LAN. To do so, simply unpack a Windows installation DVD on a Windows or Samba file share. On the PE instance, open the Windows command line and enter the command:

```
net use w: \\<samba-server>\<share> ↩
  /user:<username>
```

Next, change to the `w:` drive and to the subdirectory where you unpacked the DVD. From there, start the Windows installation with the `setup.exe` file. Again, you can automate the process with a suitable response file.

Individual PXE Configurations

Not every client that starts by PXE has to be given the menu from the default configuration file. Before PXELinux offers a client the default, it first searches for individual configuration files. The service initially checks whether a file that matches the client's MAC address exists. For example, if the client boots with a MAC address *48:2a:01:02:03:ff*, PXELinux looks for a file named `01-48-2a-01-02-03-ff` in `/var/lib/tftpboot/pxelinux.cfg/`. If the file does not exist, the program looks for configuration files that match the IP address in hexadecimal format. If a client boots with an IP of 192.168.2.201, PXELinux looks for configuration files such as `C0A802C9`, `C0A802C`, and `C0A802`, down to `C`. If these do not exist, and only then, the start service resorts to the default configuration.

Therefore, you can create individual boot files for single clients or groups. The Dnsmasq option for assigning fixed IP addresses to individual clients by `dnsmasq.conf` is useful:

```
dhcp-host=48:2a:01:02:03:ff, ↩
  192.168.2.100,infinite
```

The address needs to be outside of the initially defined DHCP pool. If you want the selected client to keep this address, simply set the lease time to `infinite`.

PXE by UEFI

Along with the Itanium processor, Intel introduced EFI as a BIOS replacement in the 1990s, lifting the BIOS's 16-bit restriction and supporting individual extensions. Starting in 2005, an alliance of various manufacturers joined forces in the Unified UEFI Forum, which has since developed into what is now no longer a proprietary specification but the open source UEFI. Another open source implementation, EDK2, is maintained by the TianoCore community and is used in the free KVM hypervisor.

UEFI itself can address disks with the GUID partition table (GPT) format and can even access file allocation table (FAT) disks without the operating system running. On Linux, the UEFI boot partition is usually found in `/boot/efi`. UEFI allows a secure boot process by executing only correctly signed files with valid keys, which has caused debate in the past – many hardware manufacturers only implement Microsoft keys on their machines, so you can only boot Windows if Secure Boot is enabled. Today, the Microsoft shim-signed UEFI first-stage bootloader is available for Linux, which builds its own secure trust chain with programs downstream (e.g., GRUB or the Linux kernel). The computer itself only has to trust the shim to boot open source Linux securely. However, users do have to create their own keys and sign the other programs in the boot sequence (e.g., GRUB and the system kernel). In the remainder of this article, I do not go further into Secure Boot but, instead, look at UEFI boot with Secure Boot disabled.

Listing 2: `grub.cfg`

```
set default="0"
function load_video {
        insmod efi_gop
        insmod efi_uga
        insmod video_bochs
        insmod video_cirrus
        insmod all_video
}
load_video
set gfxpayload=keep
insmod gzio
insmod part_gpt
insmod ext2
set timeout=10
```

Listing 3: PXELinux Example

```
menuentry ‚Fedora 34 Live‘ --class fedora --class gnu-linux --class gnu --class os {
   linux (http,<IP address of DHCP server>)/f34/images/pxeboot/vmlinuz ro rd.live.image rootovl=1
     root=live:http://<IP address of DHCP server>/f34/LiveOS/squashfs.img initrd (http,<IP address of
     DHCP server>/)/f34/images/pxeboot/initrd.img
}
```

Extending Dnsmasq

To start a designated Linux computer with UEFI, you first need the primary (shim) and secondary (GRUB) launcher. The files you need, `shimx64.efi` and `grubx64.efi`, are available from the repositories of your Linux distribution. The easiest approach is to turn to a previously installed Linux machine with a `/boot/efi` partition. Copy the two files from there into the `/var/lib/tftpboot` directory on your Dnsmasq server. On Fedora, for example, the bootloaders are located in `/boot/efi/EFI/fedora`.

Next, build a new boot menu for the UEFI machines, this time with GRUB rather than Syslinux, by first creating a `/var/lib/tftpboot/grub.cfg` file. The `grub.cfg` syntax is a bit more complex, but it is documented in great detail online [4]. In Listing 2, I used a small default header, which is followed by the menu entries that can be started by UEFI. In the PXELinux example, the menu entry for the Fedora 34 Live system would look like Listing 3.

Like PXELinux, GRUB first looks for individual configuration files. The naming scheme is:

- `grub.cfg-01-48-2a-01-02-03-ff` for a direct MAC address
- `grub.cfg-C0A802C` to `grub.cfg-C`, depending on the IP address.

If none of these files fits the bill, and again only then, GRUB serves up the menu from `grub.cfg`.

To help the Dnsmasq server distinguish between computers that use a BIOS and computers that use a UEFI boot, you need two entries in `/etc/dnsmasq.conf`. The first assigns the *efi-x86_64* label to client architecture *7* (UEFI 64-bit PC). The second entry then sets up the boot file for this architecture:

```
dhcp-match=set:efi-x86_64,➘
  option:client-arch,7
dhcp-boot=tag:efi-x86_64,shimx64.efi
```

Now the Dnsmasq server can easily handle BIOS PXE clients with PXELinux and UEFI PXE clients with the shim and GRUB.

Conclusions

Once the PXE server is up and running on the LAN, you can set up new machines and VMs quite easily with a network-based install. Both the Windows and Linux installation programs support various options for unattended setup. All you then have to do for a new system on the LAN is boot through the LAN and select the appropriate PXE entry for automatic installation. Do not forget the diskless client-server and disaster recovery use cases. If you run a small cluster with virtualization servers that use shared iSCSI or NFS SAN storage for the VMs, you can run them without hard disks and boot over the LAN. ■

Info

[1] Syntax of a Syslinux menu: [https://wiki.syslinux.org/wiki/index.php?title=Menu]

[2] Bootable WinPE media: [https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/winpe-create-usb-bootable-drive]

[3] iPXE wimboot: [https://ipxe.org/wimboot]

[4] GNU GRUB Manual 2.06: [https://www.gnu.org/software/grub/manual/grub/grub.html]

LibreOffice Expert
Dive deep into the world's greatest free office suite
2022 Edition
301 BEST BASH COMMANDS
LINUX SHELL HANDBOOK
SUPERCHARGE YOUR LINUX SKILLS
JOIN THE LINUX REVOLUTION!
GETTING STARTED WITH LINUX
MakerSpace
HANDS-ON PROJECTS FOR MAKERS
BUILD A RASP PI RADIO!
MakerSpace #02
TUNE YOUR LINUX SYSTEM
101 COOL LINUX HACKS
Tricks and shortcuts for Linux geeks
WWW.LINUX-MAGAZINE.COM

Rancher manages lean Kubernetes workloads

Construction Guide

The Rancher lightweight alternative to Red Hat's OpenShift gives admins a helping hand when entering the world of Kubernetes, but with major differences in architecture. By Martin Loschwitz

When confronted with Kubernetes, many admins immediately think of the major league products by Red Hat and Ubuntu. Red Hat's OpenShift, for example, is a massive Kubernetes distribution comprising multiple components, an integrated app store for container applications, and a great deal of complexity under the hood.

Rancher sets the bar somewhat lower. The product now belongs to SUSE, which has so far largely not interfered with the Rancher developers' work. Because of this, Rancher has retained much of the simplicity that its fans have loved for years. It also means that if you want to get started with Kubernetes today, you'll find Rancher a virtually barrier-free way of doing so.

That said, even a Rancher setup is not exactly a no-brainer. Various factors have to be considered in terms of the hardware, especially the number of machines you require and their dimensioning. Once the hardware is ready to run in your racks, the next step is to install a Kubernetes distribution, because Rancher maps its own infrastructure completely in Kubernetes. Although this sounds complicated, it is not a problem in practice because Rancher has its own Kubernetes core distribution in tow in the form of K3s to provide all the required features.

In this article, I guide you through the process of adding Rancher to your setup. Starting with a proverbial greenfield, my goal is to help you set up a Rancher cluster that can be used in production. To achieve this goal, however, I first need to clarify a few terms from the Rancher world up front with which you can assume you will be regularly confronted.

Rancher Architecture

If you already have some experience with one of the other major Kubernetes distributions, you may be a bit overwhelmed with Rancher's central features at first because Rancher works differently from most competitor products. A comparison quickly makes this clear. OpenShift, for example, consists of a management plane, or control plane, that encompasses all of the environment's core services. Although OpenShift also uses Kubernetes for its management, an OpenShift setup includes precisely one Kubernetes instance. If users launch containers in Kubernetes, the containers run on the existing cluster and use the existing infrastructure.

Rancher deviates massively from this approach. It not only sees itself as a tool for managing workloads in Kubernetes but also as a tool for controlling and managing Kubernetes environments. Therefore, applications do not run in the Kubernetes cluster that Rancher requires for its own Kubernetes components. Instead, Rancher assumes that each end-user setup has its own Kubernetes installation, which Rancher then also runs.

Because Rancher installs agents in these satellite setups, it is able to power workloads in these secondary Kubernetes clusters, too. As a consequence, the programs that a

Lead Image © Sebastian Duda, 123RF.com

Kubernetes cluster is supposed to run in Rancher never run in the same instance as the Rancher services themselves but always in a separate Kubernetes environment that Rancher controls somewhere downstream.

Some administrators might already be cringing at this point. After all, the Rancher approach seems clumsy at first glance, and above all, it seems to waste resources. It is true that the Kubernetes infrastructure components that belong to each Kubernetes cluster generate some overhead themselves, but because the individual Kubernetes instances in Rancher rarely run thousands of pods – you would run individual K8s instances for these if they're not part of the same setup – the resulting compute overhead is manageable.

On top of this, and on a positive note, you have a genuinely clear-cut separation of individual applications in the Kubernetes cluster, as well as comprehensive monitoring and alerting capabilities for each end-user Kubernetes instance. Additionally, this approach lets Rancher offer an option that other environments do not have. If required, Rancher also manages the commercial Kubernetes offerings found in AWS, Azure, or Google Cloud, for example.

Hardware Procurement

Once an organization has decided to use Rancher, the first thing on the agenda is basic deployment, and you need hardware for this. It is true that Rancher can also be completely virtualized, and, in principle, nothing can be said against this approach. However, if third-party workloads with completely different applications are running on certain nodes in addition to the Kubernetes components for Rancher, you risk resource bottlenecks.

Like any other application, Rancher feels most at home on its own hardware. Because the setup in this example is supposed to reflect a Rancher setup in a production environment in the most realistic way possible, I will be assuming that Rancher uses its own hardware. To achieve high availability, you need two servers, each with 128GB of RAM and at least 32 virtual cores (vCPUs). Unfortunately, what the manufacturer means by vCPUs in this context is not clear from the documentation.

You can safely assume that a recent Xeon with 24 physical cores (i.e., 48 threads) will be fine for most Rancher setups. Rancher itself states that a machine of this dimension can meaningfully run 2,000 clusters and up to 20,000 compute nodes, even with the database belonging to Rancher grabbing two cores and 4GB of RAM for itself in the background. Input/output operations per second (IOPS) values are more important for the database. The systems therefore ideally need to reside on fast flash memory for the servers.

In a normal setup, the worker nodes would also need at least two systems (i.e., the hosts on which the Kubernetes user clusters will run). The documentation alternately refers to these as compute or target nodes, but it means the same servers in all cases. There are virtually no limits to your imagination when it comes to hardware, but make sure the systems are not under-dimensioned. Unlike machines in full and paravirtualized setups, containers do not need CPU time to run their own vCPUs and Linux, but the containers and their applications do need to have enough RAM.

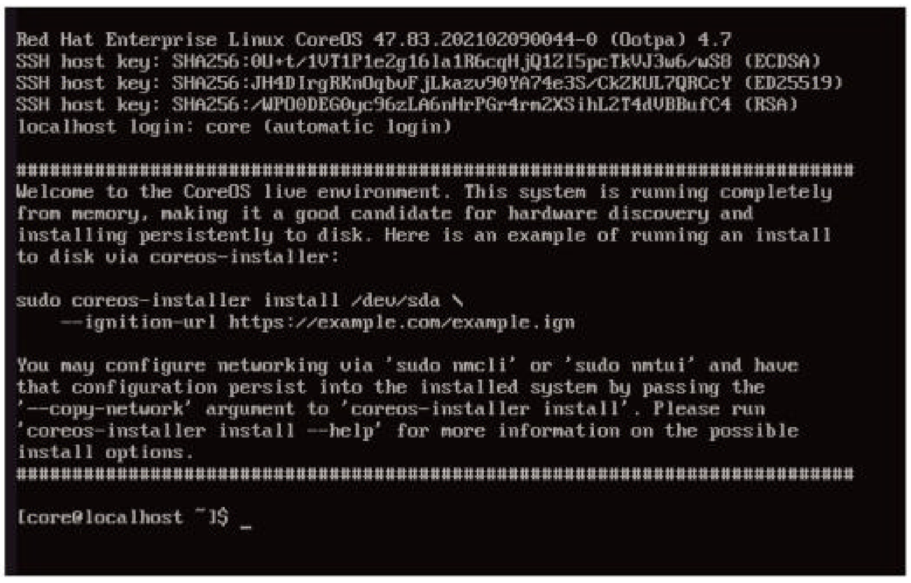
If you dimension the target nodes like nodes for operations with KVM, you normally will not put a foot wrong. Unlike the nodes for the Rancher components, you need to pay attention to local storage media on the systems for the compute nodes; after all, the containers are stored there during operation. A fast but small NVMe-based drive might endanger your setup. If you designate a few gigabytes of drive space, you are on the safe side in most cases.

Key Question

For Rancher services to run on a system at all, a Linux installation is an absolute prerequisite. This requirement forces you to opt for a specific distribution to run with Rancher. You might be tempted to use what you already know from your own experience, but doing this may be wasting a great opportunity because Rancher's components are not legacy software packages: Rancher itself is fully containerized.

The upside is that any distribution suitable for running containers with Docker or Podman is also great for Rancher. From the administrator's point of view, there is no reason to opt for a complete Linux system – a microdistribution such as CoreOS (Figure 1) is all you need. Flatcar Linux is an alternative, and one that I wrote about in the past [1]. Ubuntu Core is another option. Rancher even had its own downscaled Linux named RancherOS in the past, but it is no longer officially supported.

If you feel unhappy with a microdistribution, the usual suspects will do the job, but make sure you roll out as lean a system as possible right from the outset. On top of this, the Rancher roll-out provides a good



```
Red Hat Enterprise Linux CoreOS 47.83.202102090044-0 (Ootpa) 4.7
SSH host key: SHA256:0U+t/1VT1P1e2g16la1R6cqHjQ1Z15pcTkVJ3w6/wS8 (ECDSA)
SSH host key: SHA256:JH4DIrgRKnOqbvFjLkazv90YA74e3S/CkZKUL7QRCcY (ED25519)
SSH host key: SHA256:/WPO0DEG0yc96zLA6nHrPGr4rm2XSihL2T4dVBBufC4 (RSA)
localhost login: core (automatic login)

###########################################################################
Welcome to the CoreOS live environment. This system is running completely
from memory, making it a good candidate for hardware discovery and
installing persistently to disk. Here is an example of running an install
to disk via coreos-installer:

sudo coreos-installer install /dev/sda \
    --ignition-url https://example.com/example.ign

You may configure networking via 'sudo nmcli' or 'sudo nmtui' and have
that configuration persist into the installed system by passing the
'--copy-network' argument to 'coreos-installer install'. Please run
'coreos-installer install --help' for more information on the possible
install options.
###########################################################################

[core@localhost ~]$ _
```

Figure 1: The target systems for Rancher do not have to be kitted out with a full Linux distribution. CoreOS or Flatcar Linux are quite okay for running container workloads. © Red Hat

opportunity for taking care of automation issues. Kickstart rules for AlmaLinux or Rocket Linux can be created quickly and will save you a huge amount of work, especially when you scale the platform later. Additionally, make sure from the outset that you automate the system configuration on the individual servers to the extent possible.

One thing becomes clear at this point. A sensibly planned Rancher setup requires some work up front before it can go live, but this investment will pay dividends later on because you will be able to grow the cluster quickly and easily.

Getting Started

In this example, I have four servers running Ubuntu 20.04 LTS. Because Podman is primarily a Red Hat invention, nothing is in the way of Ubuntu continuing to use Docker as a container runtime environment. Accordingly, Docker's community repositories are activated in the system configurations, and the packages you need for Docker are installed. In this state, the systems are ready for the Rancher installation.

Terms

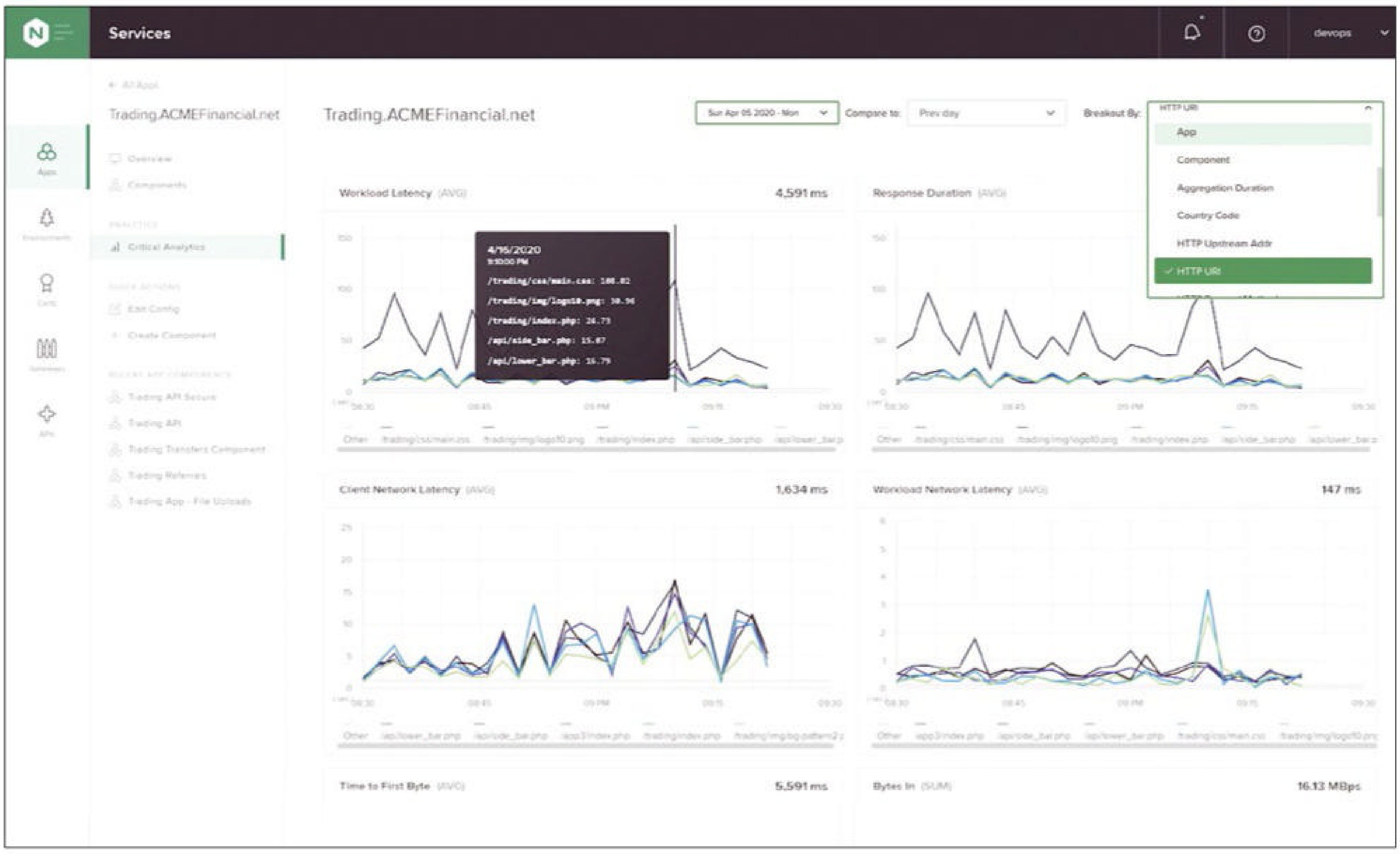
Like virtually every cloud-native solution, some terms have taken on a unique meaning in the Rancher universe over time. Before working with Rancher, you need to know the most important terms. For example, when people talk about a Rancher server, they are talking about a system that hosts all of Rancher's components for managing and provisioning Kubernetes clusters. Importantly, you must have at least one of these Rancher servers, but Rancher also allows for full scalability at its management level. As a result, the maximum number of Rancher servers per installation is virtually unlimited.

K3s is equally important in the Rancher context. This minimal Kubernetes version is maintained by the Rancher developers themselves and only contains the components you need to run Rancher. As a reminder, Rancher also rolls itself out as a Kubernetes cluster. K3s provides the foundations for this – and far leaner ones than OpenShift and others.

K3s, by the way, is not the only Kubernetes distribution that Rancher handles. You will also regularly encounter the Rancher Kubernetes Engine (RKE), a K8s distribution that comes directly from the Rancher developers but is older than K3s. The somewhat more modern K3s is therefore the recommended standard distribution for new setups. Additionally, RKE2 is a further development of RKE, which focuses on security and is designed for special use cases (e.g., in government offices).

Creating the Infrastructure

The first step en route to a working Rancher instance is installing K3s. The following example assumes that the setup has a MariaDB or MySQL database where Rancher can store its metadata. The database can optionally run on the same servers as Rancher, but an external database on its own hardware is also an option. Make sure that the database in question is highly available: Without its



4/16/2020
9:00:00 PM
/trading/css/main.css: 108.82
/trading/img/logo10.png: 30.96
/trading/index.php: 24.73
/api/side_bar.php: 15.87
/api/lower_bar.php: 15.79

Figure 2: Rancher needs a load balancer (Nginx here) to make the setup highly available. A Layer 4 load balancer is the best option for beginners. © Nginx

Listing 1: Installing cert-manager

```
# kubectl apply -f https://github.com/jetstack/cert-manager/releases/download/v1.5.1/cert-manager.crds.yaml

# helm repo add jetstack https://charts.jetstack.io

# helm repo update

# helm install cert-manager jetstack/cert-manager --namespace cert-manager --create-namespace --version v1.5.1

# kubectl get pods --namespace cert-manager
```

metadata, Rancher is more or less useless. MariaDB or MySQL must run reliably.

Because the Rancher server components in this example use a redundant design, you will need a load balancer. Ideally, the load balancer will listen for incoming connections on the address that Rancher uses later; it forwards these connections to two Rancher hosts. Load balancers on protocol Layer 4 and on Layer 7 are possible.

I assume that a Layer 4 load balancer is used (Figure 2), which makes the setup a little easier, especially with a view to Rancher's Secure Sockets Layer (SSL) capabilities. A Layer 7 device would offer more configuration options in theory, but you would need to configure it to handle SSL management. If you have a Layer 4 load balancer, Rancher handles SSL management itself, rolling out an instance of Traefik [2] to do so.

Either way, the load balancer must also be highly available. Its failure would mean that Rancher itself and the managed Kubernetes clusters might still be running, but they would be inaccessible from the outside.

Speaking of accessibility: The DNS entry intended for Rancher needs to be added to the zone file for your domain when you set up the load balancer; otherwise, you can't start to install Rancher itself.

Last but not least, the time needs to be correct on every system of the installation and can be implemented within the distributions either by the legacy network time protocol (NTP) or with chronyd. Either way, one of the two components needs to be set up and have access to an NTP server to set the system time correctly.

Rolling Out K3s

The K3s authors make it very easy for you to install their software on your system. To do so, simply run a command in the form

```
$ curl -sfL https://get.k3s.io | ⏎
  sh -s - server --datastore-endpoint=⏎
    "mysql://<User>:<Password>⏎
     @tcp(<Host>:3306)/<Database>"
```

on each machine intended as a Rancher server, replacing `<User>`, `<Password>`, `<Host>`, and `<Database>` with your MySQL database credentials. Assuming the username *rancher*, password `secret`, and database name `rancher`, the command would be:

```
$ curl -sfL https://get.k3s.io | ⏎
  sh -s - server --datastore-endpoint=⏎
    "mysql://rancher:secret⏎
     @tcp(10.42.0.1:3306)/rancher"
```

Of course, the command specified here will only work if the system has direct access to the Internet. This is not absolutely essential for Rancher because the software can also use a proxy server or run without a network connection in air gap mode. However, describing these installation variants is beyond the scope of this article, so check out the Rancher documentation [3] if you need these options.

Checking K3s

On all future Rancher servers, after successfully completing the installation, the

```
sudo k3s kubectl get nodes
```

command should return a list of all Rancher servers in the setup. If the list shows both machines (in this example), the K3s setup worked.

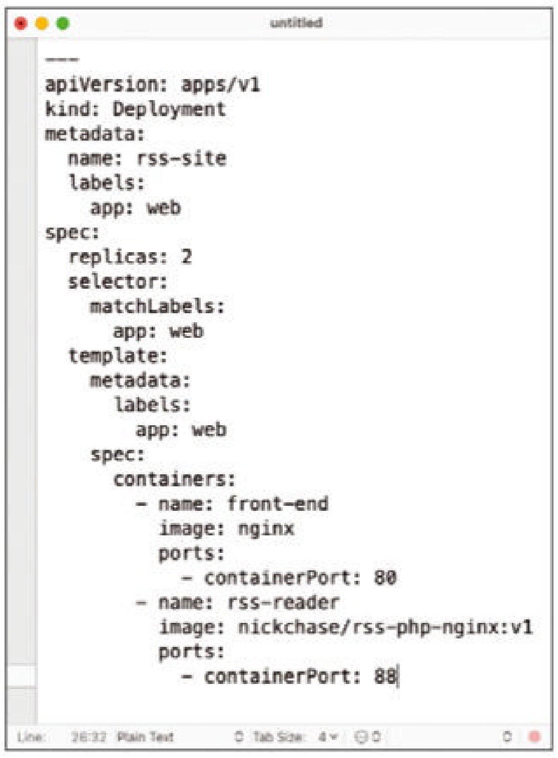
Of course, the K3s tool is K3s-specific. It would be useful to be able to access the K3s cluster with the standard `kubectl` tool, too. So that you can do so, K3s created an `/etc/rancher/k3s/k3s.yaml` file during the installation, which every user with execute rights will want to copy to `~/.kube/config`. Before that, you have to adapt the file because the host to be managed defaults to `localhost`. Find the `server` entry in the YAML file and replace this value with the DNS name pointing to the load balancer mentioned earlier. After doing so, the

```
kubectl get pods --all-namespaces
```

command should work. If so, K3s is now ready for the Rancher installation.

Installing the Certificate Manager

Because I want Rancher to pick up its SSL certificates automatically by Let's Encrypt, I now need to install `cert-manager`. Several commands let you do this. The command in the first line of Listing 1 installs the custom resource definitions (CRDs) required for `cert-manager` in the local



```
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: rss-site
  labels:
    app: web
spec:
  replicas: 2
  selector:
    matchLabels:
      app: web
  template:
    metadata:
      labels:
        app: web
    spec:
      containers:
        - name: front-end
          image: nginx
          ports:
            - containerPort: 80
        - name: rss-reader
          image: nickchase/rss-php-nginx:v1
          ports:
            - containerPort: 88
```

Figure 3: The Helm package manager for Kubernetes works in a similar way to the established package managers for RPM and DPKG. It bundles image metadata and images and makes them downloadable as a whole.

Listing 2: Rancher Install

```
# helm repo add rancher-latest https://releases.rancher.com/server-charts/stable
# kubectl create namespace cattle-system
# helm repo update
# helm install rancher rancher-latest/rancher --namespace cattle-system --set hostname=<I>Host<I>
  --set replicas=3 --set ingress.tls.source=letsEncrypt --set letsEncrypt.email=<I>E-Mail<I>
  --set letsEncrypt.ingress.class=nginx
# kubectl -n cattle-system rollout status deploy/rancher
# kubectl -n cattle-system get deploy rancher
```

K3s instance, and the command in the second line adds the Helm directory for `cert-manager` to the K3s instance.

The command in line 3 updates the local metadata of all configured Helm directories before the command in line 4 drops `cert-manager` into the local K3s installation. If everything works, the call in the last line displays the running containers for `cert-manager`.

Installing Rancher

The Rancher developers also distribute their product as a Helm chart. As a reminder, the Helm package manager (Figure 3) for Kubernetes lets you deliver metadata and images in a standardized format. Like the distributions' normal packages, Helm charts can be obtained from different directories.

For Rancher, first add the Helm directory of the Rancher project to your K3s installation (Listing 2, line 1). Then create a namespace in K3s in which all the Rancher services will run (line 2) and update the metadata of the available Helm charts again (line 3). The command in line 4 creates a running Rancher cluster, where you need to replace `<Host>` and `<Email>` with the values for your use case.

The command in line 5 shows the progress of the install, which can take some time. The final command should show a ready-to-use deployment with three instances available. The Rancher installation is now complete.

If you now open the URL that contains the address of the load balancer, you will automatically be taken to the installation's login page. Rancher displays the user data at the command line during the deployment process, but the first time you log in to the web interface, you will need to change the credentials. Use the command

```
# kubectl get secret ↩
  --namespace cattle-system ↩
            bootstrap-secret ↩
  -o go-template=↩
    '{{ .data.bootstrapPassword|↩
        base64decode}}↩
     {{ "\n" }}'
```

if you did not make a note of the password during the installation.

Adding Nodes

Although you currently have a usable Rancher cluster, you can't roll out workloads on it yet. The nodes – the systems that run the Kubernetes clusters for the applications – are missing. In terms of preparation, the same steps apply to the nodes as to the Rancher servers, except you do not need a database; however, NTP must again be active.

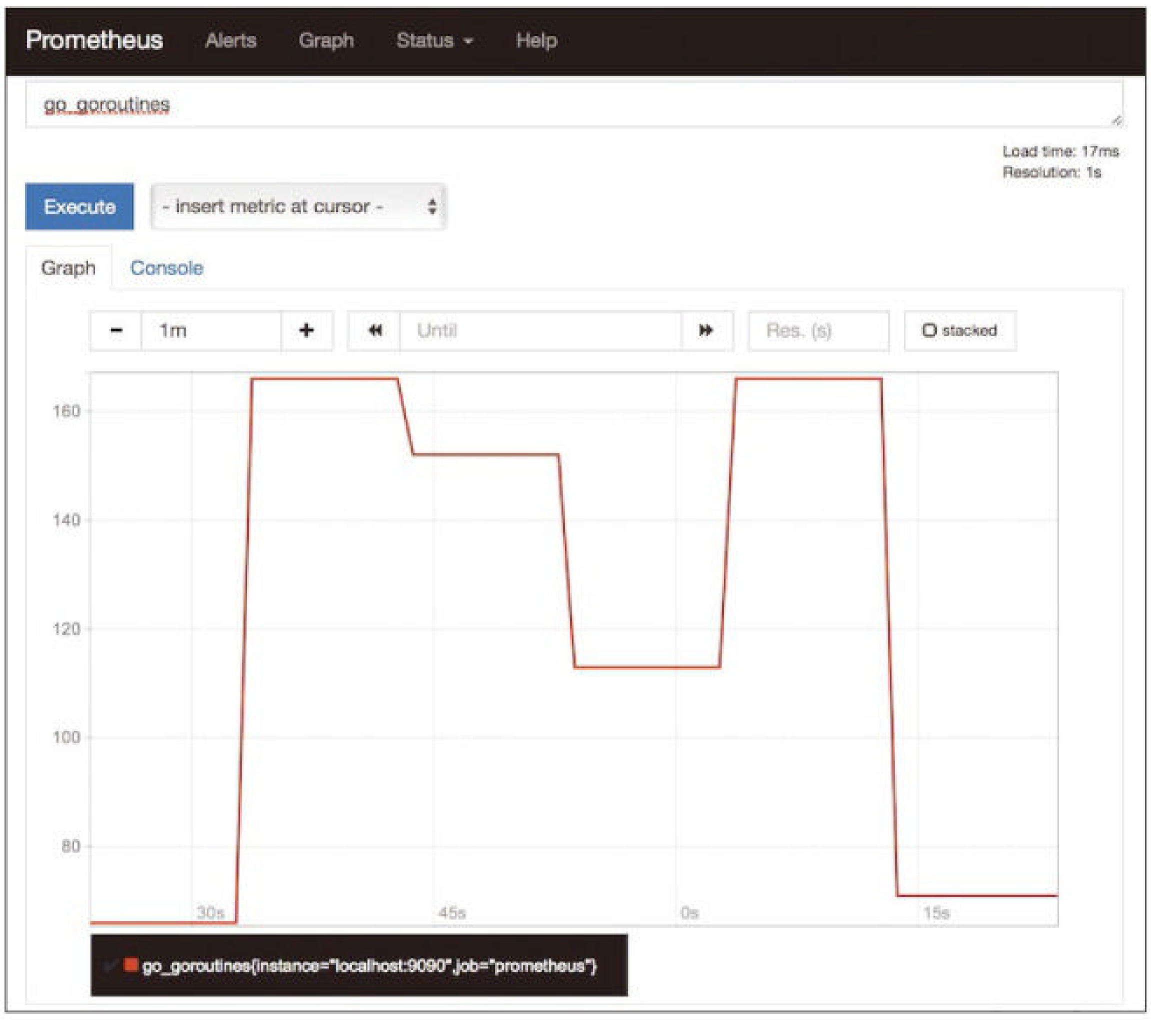
Once all the requirements are met, the command

```
# curl -sfL https://get.k3s.io | ↩
  K3S_URL=https://<Rancher-Hostname>:6443 ↩
  K3S_TOKEN=<Token> sh --
```

is all it takes. Replace `<Rancher-Hostname>` with the hostname for the load balancer. The content you need to specify for `K3S_TOKEN` can be found on the servers in the `/var/lib/rancher/k3s/server/node-token` files.

Using Rancher

Immediately after taking these steps, the first workloads can be rolled out



go_goroutines

Load time: 17ms
Resolution: 1s

go_goroutines{instance="localhost:9090",job="prometheus"}

Figure 4: Prometheus is a very powerful tool for collecting metrics. It runs under the hood in Rancher to provide data from Rancher and the end-user setups running there. © Alex Eillis

to Rancher. You can also access the Rancher Marketplace from the web user interface, which is where preconfigured applications optimized for Rancher are available. They again find their way into the installation by Helm charts.

Of course, this far from exhausts Rancher's capabilities. Once the first applications are rolled out as services in Rancher, you can set up monitoring for them from the *Monitoring* menu item, including alerting on the basis of various parameters. If this combination of components looks familiar to you from the context of cloud-native environments, you are on the right track, because Rancher does not implement monitoring itself. Instead, it relies on a combination of Prometheus (Figure 4), the matching Alert Manager, and the Grafana GUI component (Figure 5) in the background.

Conclusions

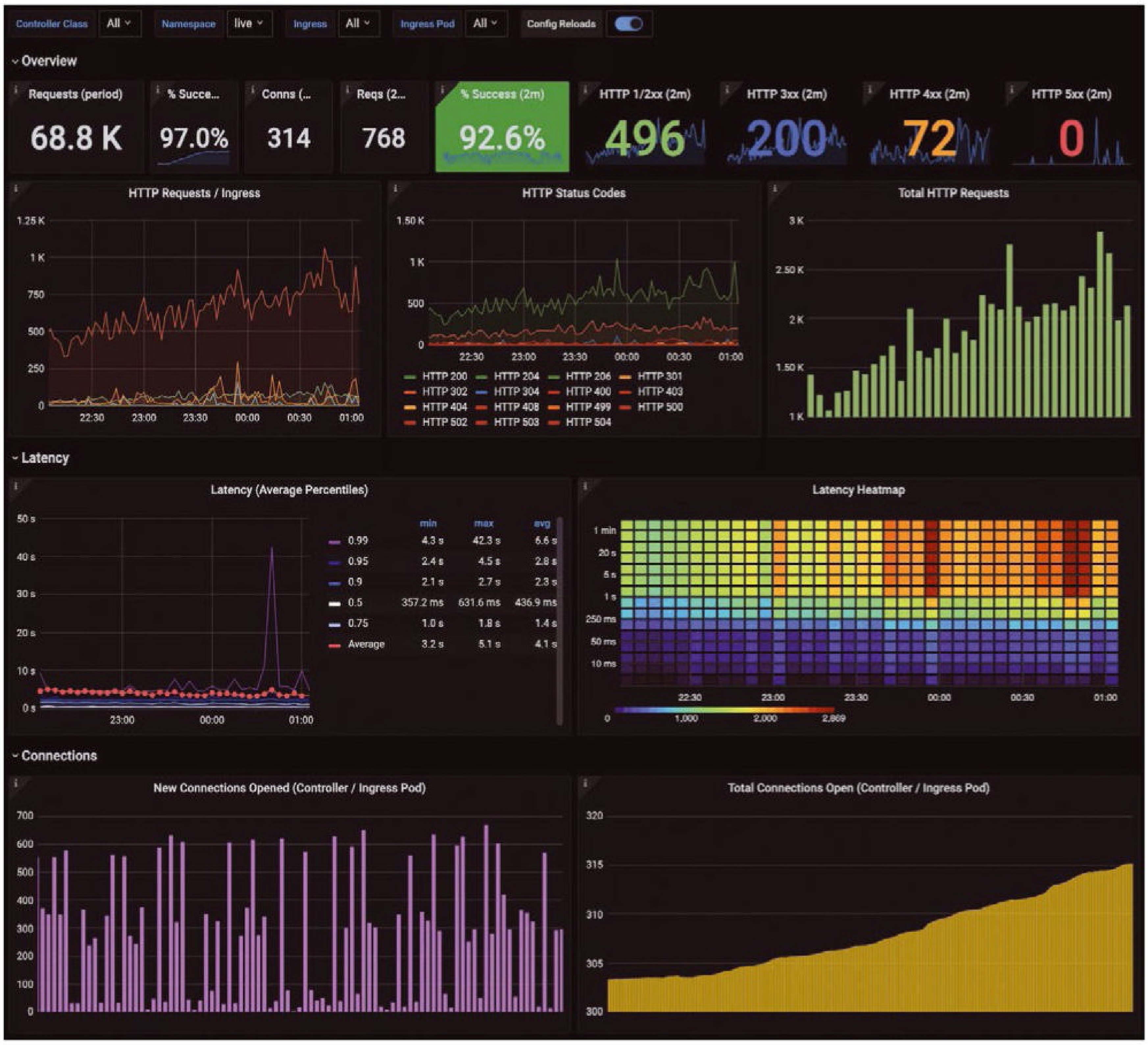
Rancher does impose a number of requirements in terms of the infrastructure it expects to have in place, but once rolled out, it turns out to be an extremely powerful tool for running Kubernetes workloads. The setup turns out to be an intuitive and comparatively smooth process, so if you are looking to deploy Kubernetes, you may well want to include Rancher in your evaluation process. ■

Info

[1] "Container Microdistributions k3OS and Flatcar" by Martin Gerhard Loschwitz, *ADMIN*, 2020, issue 60, pg. 24, [https://www.admin-magazine.com/Archive/2020/60/Container-microdistributions-k3OS-and-Flatcar/]

[2] "Managing Network Connections in Container Environments" by Martin Gerhard Loschwitz, *ADMIN*, 2021, issue 63, pg. 58, [https://www.admin-magazine.com/Archive/2021/63/Managing-network-connections-in-container-environments/]

[3] Rancher docs: [https://rancher.com/docs/]



Controller Class: All · Namespace: live · Ingress: All · Ingress Pod: All · Config Reloads

	min	max	avg
0.99	4.3 s	42.3 s	6.6 s
0.95	2.4 s	4.5 s	2.8 s
0.9	2.1 s	2.7 s	2.3 s
0.5	357.2 ms	631.6 ms	436.9 ms
0.75	1.0 s	1.8 s	1.4 s
Average	3.2 s	5.1 s	4.1 s

Figure 5: The combination of Prometheus, the matching alert manager, and Grafana is also available, which, configured appropriately by Rancher, helps monitor container workloads. © Grafana

Five free wiki platforms reviewed

Private Library

If you want to operate a wiki in your data center as a document and knowledge database for workgroups, you will find a large selection of open source applications. We tested five free wiki platforms for different use cases on the local network. By Andreas Stolzenberger

Wikipedia is famous not just as a large encyclopedia; the web software also helps users collaborate on online documents. Wiki markup (or wikicode or wikitext) lets you format documents with very simple methods and without the need to use special WYSIWYG editors or complex page description languages, as in HTML, XML, and LaTeX (for technical and scientific documents). Therefore, wikis are often used on local networks for documentation and internal document collections for code repositories, project management tools, and IT service management tools.

The requirements for a wiki server are fairly meager. In most cases, you just need a web server and a database server. Some systems can even manage without a database. The learning curve for users is also low because the Markdown syntax for text can be learned in next to no time. Additionally, modern wikis increasingly include optional online WYSIWYG editors that make it even easier to get started.

In this article, I look at five free wiki platforms and discuss the advantages of the different platforms. Check out the "Test Setups with Podman and Ansible" box to see how I roll out the wiki applications and databases.

DokuWiki

DokuWiki [1] is one of the most popular wiki servers around, especially for smaller installations. The PHP program does not need a separate database server; in fact, it only needs a Linux installation with an Apache or NGINX server. DokuWiki stores all pages in separate TXT files in the original format, which makes it easy to back up the complete wiki and version it with a repository management tool like Git. DokuWiki has a number of plugins that enhance both functionality and appearance.

After the initial installation, DokuWiki lets all users create and edit documents, but the tool does have granular user and group management. As the admin, you use access control lists to define which users and groups are authorized for specific areas of the wiki. Often, unauthenticated users are only allowed to read the wiki's content, whereas registered users and groups can edit within their respective subdirectories. Special pages like `start` or `sidebar` support navigation within the wiki. Users create link and menu pages in the wiki language, too. In

Test Setups with Podman and Ansible

For this article, I roll out the wiki applications and databases as containers with Ansible playbooks on a Podman host. To prevent multiple parallel installations fighting for the web or MySQL port on a single IP address on the container host, I set up the containers with their own IP addresses. To do this, I first need to define the **`host_local`** bridged network of the **`macvlan`** type on the Podman host, which means the containers do not have to run on the IP address of the host but can use their own IP addresses. Next, I create matching containers in Ansible, as this excerpt from a playbook shows:

```
...
- name: wiki.js
  containers.podman.podman_container:
    name: wikijs
    image: docker.io/requarks/wiki:2
    state: started
    network:
    - host_local
    ip: "192.168.2.188"
    dns: "192.168.2.2"
    expose:
    - "3000"
    env:
....
```

Photo by Mariia Zakatiura on Unsplash

addition to internal user management, DokuWiki can connect to authentication services such as Active Directory (AD), Lightweight Directory Access Protocol (LDAP), and others.

The built-in media manager takes care of additional binary files (e.g., images, movies, PDFs) that can be linked into the wiki pages. DokuWiki's initial look is very functional, but a bit outdated. Fortunately, you can freshen up the look with themes. That said, the selection is limited and cannot keep pace with the theme collection of a blog system like WordPress. Plugin and theme management is well programmed. As the administrator, you can search for extensions directly in the administration menu and install them right away.

Ready Fast

The wiki is simple and ready to use in a few minutes. Anyone looking to come to grips with the topic of wiki servers, but with no previous experience, would do well to start with a DokuWiki installation (Figure 1).

As the administrator, you write the complete wiki structure and menu navigation, just like documents, in the Markdown "WikiCloth" syntax, which is also used by MediaWiki (Wikipedia). Simple themes like *Bootstrap3* spice up the look of the wiki but keep the functionality aspect in the foreground. Managing and saving the content is very easy, because DokuWiki does not require a database. One downside for beginners, however, is that DokuWiki only uses WikiCloth and does not offer a WYSIWYG editor. If you frequently work with GitHub and would like to stay with the Markdown used there, DokuWiki has a number of Markdown plugins that support Markdown in addition to WikiCloth. The *stackedit* plugin, in combination with *mdpage*, adds a very useful WYSIWYG Markdown editor to the DokuWiki system.

Wiki.js

The JavaScript Wiki.js [2] application was released at the end of 2016 and has grown by leaps and bounds in recent years. Wiki.js combines original wiki functionality with an appealing, modern user interface (UI) and an integrated WYSIWYG editor for documents, making it far easier for users without Markdown experience to get started. If you are familiar with Markdown, you will be pleased to know that Wiki.js does not use its own dialect; instead, it relies on the standard used by GitHub.

Wiki.js also offers a flexible plugin system for expanding the feature set. Features include renderers (i.e., alternative page description languages). If you want to use technical and scientific documents in Wiki.js, you will be happy to find a LaTeX renderer. Modules also ensure that user authentication with back-end services such as Google, LDAP, or similar is supported.

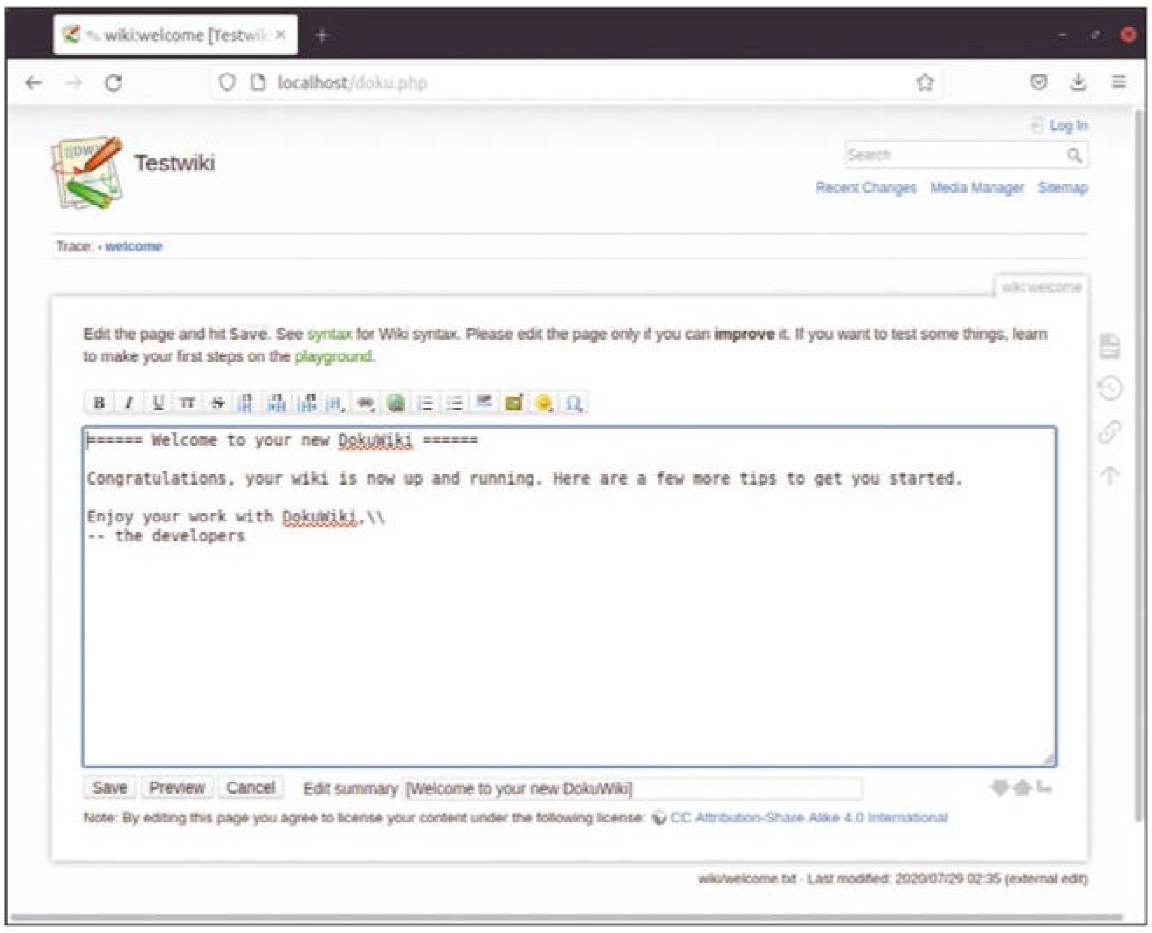
The basic installation is not as simple as with DokuWiki. Wiki.js needs a PostgreSQL database, although the current version also supports back ends such as Microsoft SQL, MariaDB, or even plain vanilla SQLite. Support for these databases is on the scratch list, though, so future versions will only work with PostgreSQL [3].

In the test setup, you just need two containers: one for the wiki app and one for the PostgreSQL database. The wiki container does without a passthrough directory for storing data on the host disk. All information is stored in the database without a separate document directory.

The application comes with support for a number of file back ends like Amazon Web Services Simple Storage Service (AWS S3), Secure File Transfer Protocol (SFTP), or Git, although they do not serve the live application, but provide backup storage. When running, Wiki.js has a familiar-looking state-of-the-art web interface, although some processes are more complicated than with the simple DokuWiki. Generally speaking, many functions on the admin back end of Wiki.js still wear a "Coming Soon" tag (Figure 2). The driving company behind Wiki.js, requarks.io, has announced the new version 3 for 2022, which should see a number of improvements and changes [4].

Range of Functions

Wiki.js is an attractive, state-of-the-art application with a modular architecture. Some of the modules on the roadmap, such as flexible theme management, are still missing here



Edit the page and hit Save. See syntax for Wiki syntax. Please edit the page only if you can **improve** it. If you want to test some things, learn to make your first steps on the playground.

```
====== Welcome to your new DokuWiki ======

Congratulations, your wiki is now up and running. Here are a few more tips to get you started.

Enjoy your work with DokuWiki.\\
-- the developers
```

Edit summary [Welcome to your new DokuWiki]

Note: By editing this page you agree to license your content under the following license: CC Attribution-Share Alike 4.0 International

wiki/welcome.txt · Last modified: 2020/07/29 02:35 (external edit)

Figure 1: DokuWiki uses the same Markdown language as Wikipedia. Authors can work in a simple text editor. Plugins can also be used to retrofit a WYSIWYG editor for Markdown.

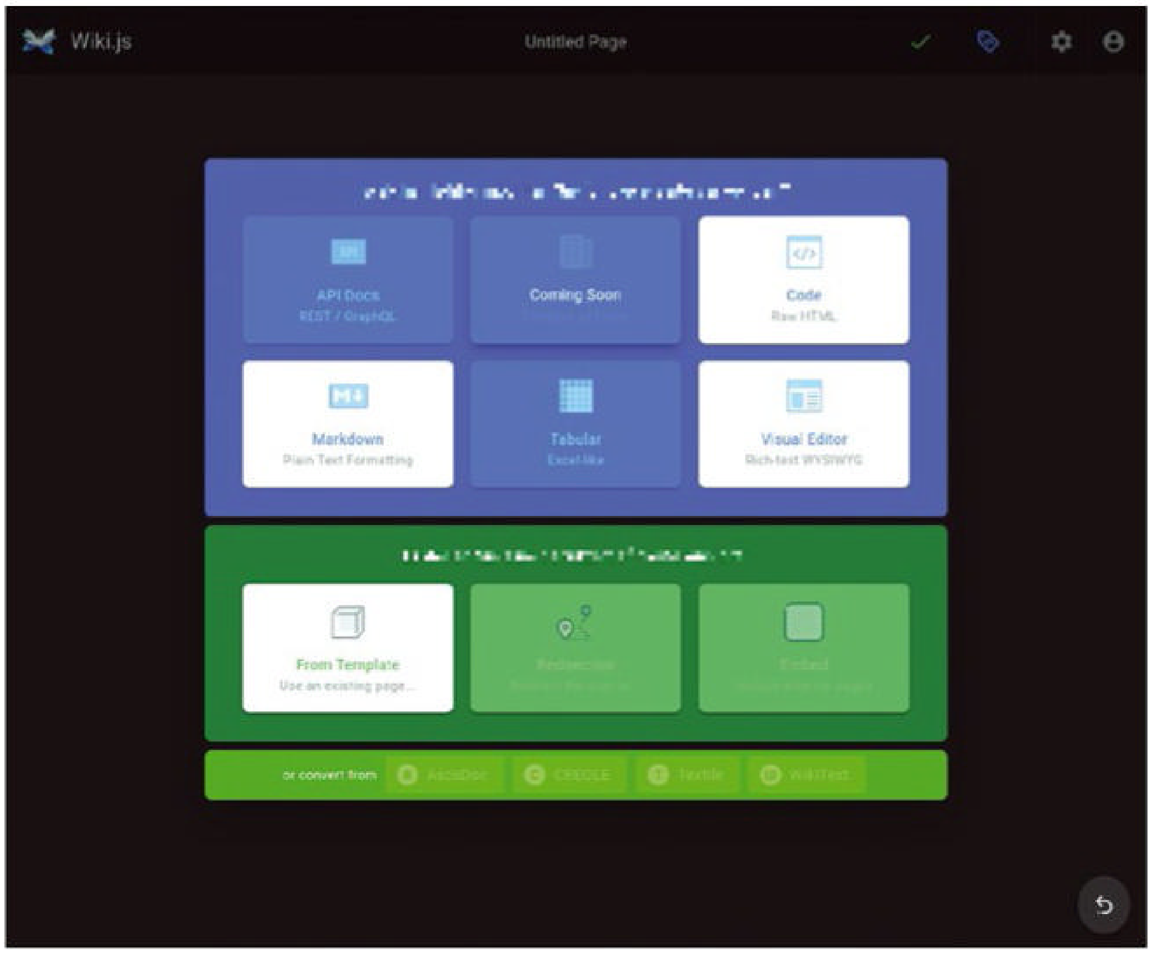


Figure 2: Wiki.js comes with a modern interface and many plugins. However, more than half of the displayed functions are due to follow in future releases.

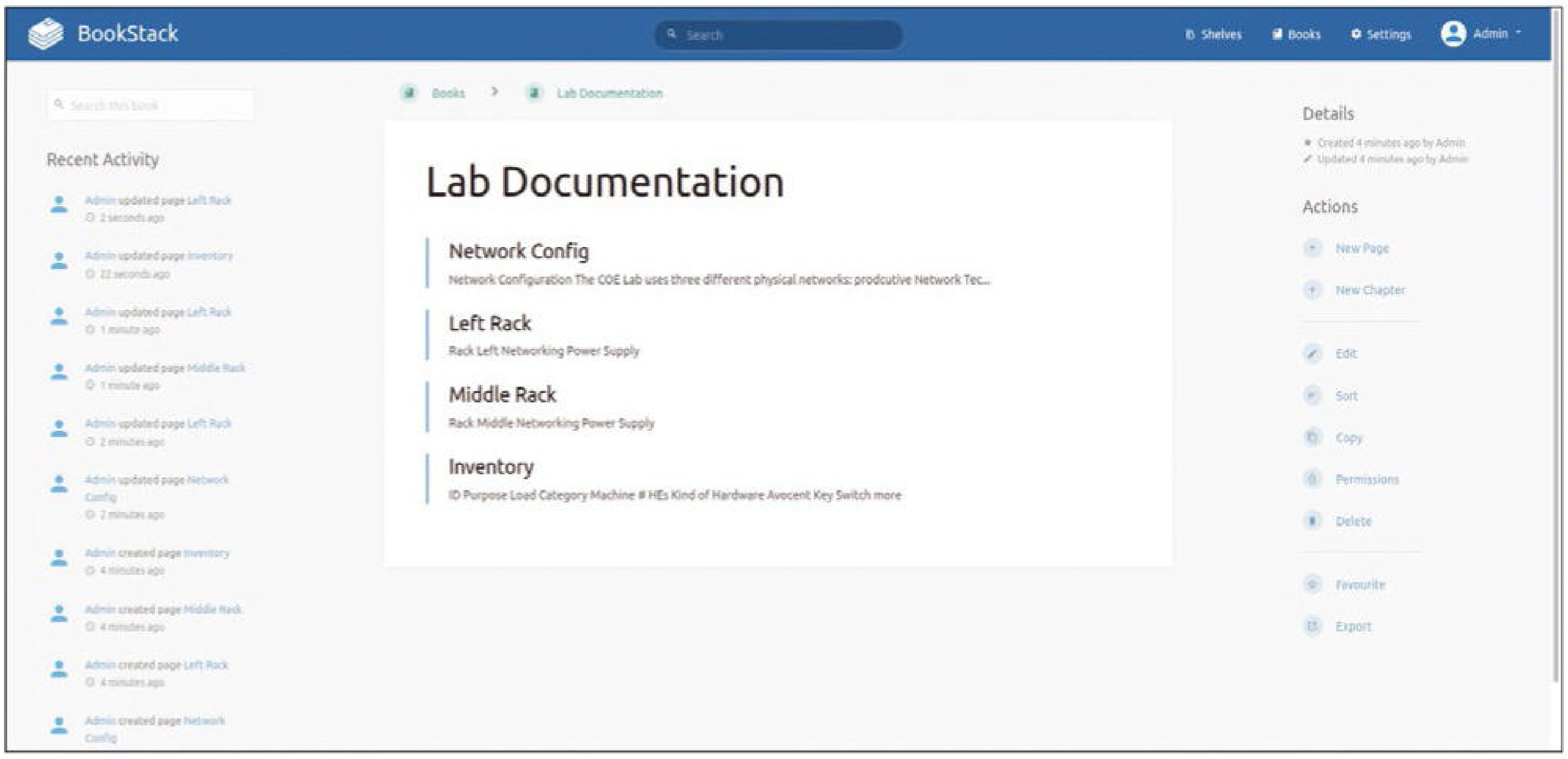
and there, so Wiki.js is still tied to a standard look. To compensate, the functional range is impressive, especially on the back end, and users get a solid and well-thought-out product. Wiki.js primarily appeals to users who want a collaborative wiki platform but prefer to work with WYSIWYG rather than Markdown. Regretfully, the editor stores the documents internally in HTML code, which means users cannot use different approaches to working on a document (e.g., Markdown and WYSIWYG).

MediaWiki

For the sake of completeness, I need to introduce MediaWiki [5] – the platform on which Wikipedia runs. In principle, anyone interested in doing so can download and use the open source code. MediaWiki requires a MySQL database. Containerized installation is also supported, with a minor workaround. MediaWiki is probably not the first choice for administrators of small to medium-sized networks. The tool is designed for very large environments and huge databases – as is the case with an encyclopedia like Wikipedia. The tool is simply too complex and confusing to manage for small and medium-sized workgroups.

BookStack

The open source BookStack [6] wiki describes itself as "an open source alternative to Confluence" [7] and sorts documents like bookshelves, with a hierarchy of Shelves, Books, and Pages. At first glance, BookStack does not look like a conventional wiki, but more like a simple document management system (DMS). The GUI is very tidy, focusing on the essentials, which also applies to the simple admin menu. Unlike tools such as DokuWiki, BookStack does not show you every imaginable configuration option in the admin menu. To set up features like LDAP integration or an S3 back end, you have to edit the ENV configuration file manually. No prebuilt themes are on offer, either, although the documentation at least points out where you can make adjustments to the stylesheet (CSS).



Recent Activity

Admin updated page Left Rack
2 seconds ago

Admin updated page Inventory
22 seconds ago

Admin updated page Left Rack
1 minute ago

Admin updated page Middle Rack
1 minute ago

Admin updated page Left Rack
2 minutes ago

Admin updated page Network Config
2 minutes ago

Admin created page Inventory
4 minutes ago

Admin created page Middle Rack
4 minutes ago

Admin created page Left Rack
4 minutes ago

Admin created page Network Config

Lab Documentation

Network Config
Network Configuration The COE Lab uses three different physical networks: prodcutive Network Tec...

Left Rack
Rack Left Networking Power Supply

Middle Rack
Rack Middle Networking Power Supply

Inventory
ID Purpose Load Category Machine # HEs Kind of Hardware Avocent Key Switch more

Figure 3: BookStack uses a clear-cut hierarchy in a simple and clear UI. The tool also provides the best WYSIWYG editor of the tested programs.

Users will quickly find their way around the tidy UI (Figure 3). Only the hierarchy can be confusing at first: The concept of managing Chapters and Pages as separate documents within a Book takes some getting used to. A Page alone could contain a complete book.

One of the strengths and weaknesses of BookStack is the editor. BookStack comes with two editors: one for WYSIWYG and one for Markdown. Both do an excellent job, and the Markdown variant offers a live preview. Users of the WYSIWYG editor can copy and paste formatted text from sources like LibreOffice or Google Office, or even from within spreadsheets, and let BookStack take care of the formatting, which is a great feature, especially if you want to populate your new BookStack wiki with existing documents. However, you can only select one of the two editors for the entire wiki. An option in BookStack like that in Wiki.js would be desirable to let users decide which editor to use for each document.

Simple

As a PHP application, BookStack is also quickly installed or launched as a container. The wiki works with the usual suspects (e.g., a MySQL or MariaDB database as the back end). Compared with other wikis, BookStack offers less flexible customization options and plugins. To compensate, the tool comes with a very simple and clear-cut interface, is very fast, and offers great editors. The environment is well suited for use as a documentation server in an internal network.

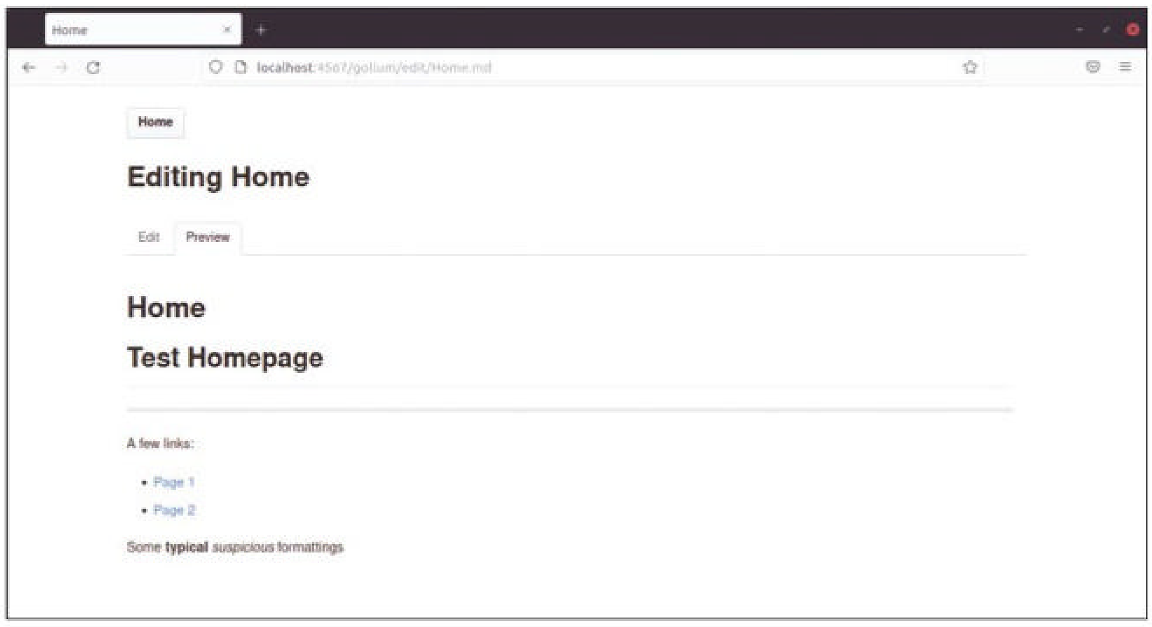
Gollum

A wiki does not always require servers, databases, and an Internet connection. Those with a need to prepare, edit, and correct pages on the go with no connection to their wiki can do so with Gollum [8] (Figure 4). The small Ruby tool builds a wiki exclusively from files in a local directory and relies on Git for version control. In principle, a set of local Markdown files can also be edited with a suitable editor such as Visual Studio Code or Atom. Gollum charms users with its ability to render pages so that Markdown hyperlinks and, in turn, the wiki page structures work.

Besides regular Markdown (e.g., as on GitHub), Gollum supports other language plugins like WikiCloth or AsciiDoc. I don't have much to say about the installation: On a system with Ruby (and *ruby-devel*) in place, you just need to type

```
gem install gollum
gollum </path/to/wiki>
```

to install and start the tool, having previously run `git init` in the same path. The directory either contains existing Markdown files, or you can add content from within Gollum. The tool listens on *localhost:4567* and can be used with any browser.



Editing Home

Home

Test Homepage

A few links:

- Page 1
- Page 2

Some **typical** suspicious formattings

Figure 4: It doesn't get any simpler than Gollum. The local tool is great for preparing and editing Markdown pages offline.

Gollum has a simple Markdown editor and renderer. The tool is great for preparing Markdown content on a local computer and uploading it to a wiki later.

Conclusions

Admins who want to set up their own wiki platform have several interesting alternatives from which to choose. Of course, the list here does not claim to be complete, because a plethora of other tools exist (e.g., XWiki and Outline). The tools I looked at in this article (except MediaWiki) are easy to install and use, which means that users who are not yet familiar with the strategy, operations, and use of wikis can familiarize themselves with the material and then take a closer look at various aspects. DokuWiki exhibits the "classic" wiki structure, BookStack is the DMS among wikis, and Wiki.js is a modern architecture.

In principle, wiki platforms offer a comparatively simple approach to storing document collections centrally and making them accessible to all users in the data center. Although this setup might go against the current "cloud first" trend, not all information and documentation is necessarily better off stored on cloud servers than in your own back yard. ■

Info

[1] DokuWiki: [https://www.dokuwiki.org/dokuwiki]

[2] Wiki.js: [https://js.wiki]

[3] Wiki.js database back end: [https://blog.js.wiki/news/2021/wiki-js-3-going-full-postgresql]

[4] Wiki.js v3 feature preview: [https://blog.js.wiki/news/2022/wiki-js-3-feature-preview-storage-delivery-paths]

[5] MediaWiki: [https://www.mediawiki.org/wiki/MediaWiki]

[6] BookStack: [https://www.bookstackapp.com]

[7] BookStack as an alternative to Confluence: [https://www.bookstackapp.com/about/confluence-alternative/]

[8] Gollum: [https://github.com/gollum/gollum]

Tools for managing AWS cloud services

The Right Button

The AWS Management Console, command-line tools, SDKs, toolkits for integrated development environments, and automation through Infrastructure as Code are all tools to help manage the operation of more than 200 services in the AWS cloud. By Christopher Henkel

The obvious starting point in managing services in the Amazon Web Services (AWS) cloud is the AWS Management Console. In the first part of this article, I look at the user interface and how to set the language, harden the root user account, change the region, and create a support request. Next, I'll show you how to learn about, find, and test services and create a list of favorites. Before moving on to other tools in the second part of this article, I'll show you how to manage your environment by managing user roles and access, monitoring system health, and viewing bills and costs.

Getting Started

To log in to the Management Console user interface, go to the AWS sign-in page [1] and enter your AWS account credentials. To add another level of security, you can set up multifactor authentication (MFA). By default, a session expires automatically after 12 hours. To continue, just press the *Click login to continue* button and log in again. You can also set your own time limits for sessions according to your organization's specifications.

The console supports the three latest versions of Google Chrome, Mozilla Firefox, Microsoft Edge, and Apple Safari, as well as Microsoft Internet Explorer 11. After logging in, you will see the AWS Management Console home page (Figure 1). In addition to accessing the web, you can view your resources, including Amazon CloudWatch alerts, with the AWS console mobile app. You can also handle operational tasks on an iOS or Android mobile device, if so desired.

The display language for the AWS Management Console can be changed in any area of the console. Twelve languages are currently supported. To change the console language, log in to the AWS Management Console and select the Language menu on the left in the bottom navigation bar. When you get there, set the language you want to use.

Securing the Root User Account

Do not use the root user account for everyday tasks, including administrative tasks, under any circumstances. Only use your root user account to create an admin user; then, securely lock away the root user's credentials and use them only for a very limited selection of account and service management tasks. Roles let you delegate permissions for day-to-day tasks. AWS Single Sign-On (SSO) and AWS Identity and Access Management (IAM) support common open identity

Photo by elnaz asadi on Unsplash

standards – for example, Security Assertion Markup Language (SAML) 2.0, OpenID Connect, and OAuth 2.0 – to facilitate identity federation.

To secure your root user account, change the password and set up multifactor authentication as follows:

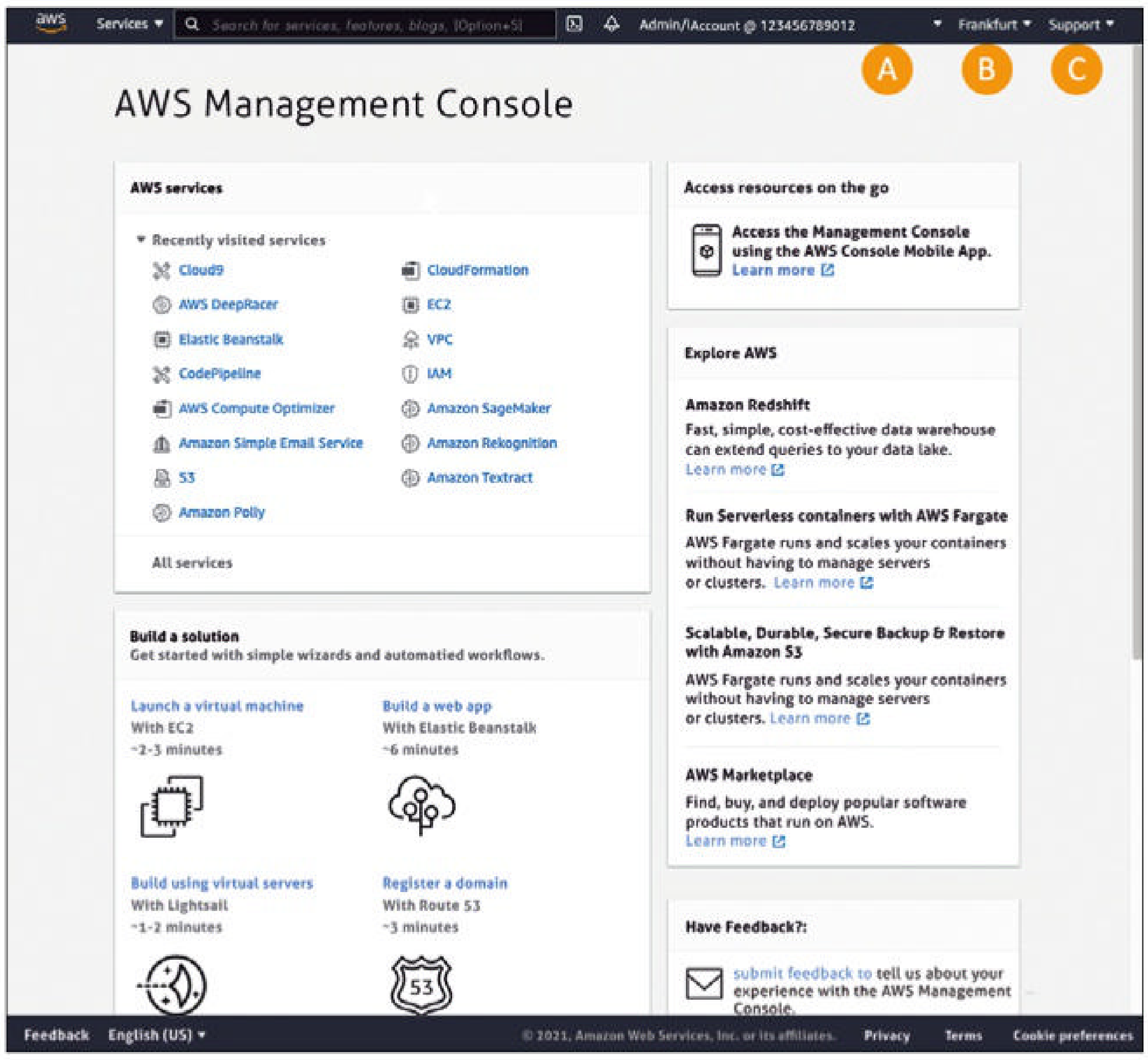
- Use the email address and password for your AWS account to log in to the AWS Management Console with the root user account (not the IAM user).
- In the top right corner of the console, select your account name or number (Figure 1A) and select *My Security Credentials*.
- Expand the Password section and press the *Click here* text to change your password. Enter your current password once and the new password twice. The new password must be at least eight characters in length and contain a symbol, a number, an uppercase letter, and a lowercase letter.
- When you have completed the password form, select *Change password* or *Save changes*.

To enable an MFA device for your regular user account, follow the first two steps from the previous password change guide; then, expand the MFA section. Select *Manage MFA* or *Activate MFA* and follow the instructions to reflect the key type: Virtual MFA Device, U2F Security Key, or Hardware MFA Device.

Regions and Support Requests

For many services, you can select an AWS region [2] that determines where your resources are managed. For some services, such as AWS IAM, no region is selectable because they are global services that you can use in any AWS region. To select a region, go to the name of the currently displayed region in the navigation bar (Figure 1B) and find the region to which you want to switch, which becomes the default in the console.

AWS Support offers a range of plans that provide access to tools and expertise that support the operational health of your AWS solutions. All support plans – including the free Basic plan – give you access to customer service, AWS documentation, technical documentation, and support forums at any time. Help with billing and account questions and requests for service limit increases are also available in all support plans.


aws
Services ▾
Search for services, features, blogs, (Option+S)
Admin/IAccount @ 123456789012 ▾
Frankfurt ▾
Support ▾
A
B
C

AWS Management Console

AWS services

▾ Recently visited services

Cloud9
AWS DeepRacer
Elastic Beanstalk
CodePipeline
AWS Compute Optimizer
Amazon Simple Email Service
S3
Amazon Polly
CloudFormation
EC2
VPC
IAM
Amazon SageMaker
Amazon Rekognition
Amazon Textract

All services

Build a solution
Get started with simple wizards and automatied workflows.

Launch a virtual machine
With EC2
~2-3 minutes

Build a web app
With Elastic Beanstalk
~6 minutes

Build using virtual servers
With Lightsail
~1-2 minutes

Register a domain
With Route 53
~3 minutes

Access resources on the go

Access the Management Console using the AWS Console Mobile App.
Learn more

Explore AWS

Amazon Redshift
Fast, simple, cost-effective data warehouse can extend queries to your data lake.
Learn more

Run Serverless containers with AWS Fargate
AWS Fargate runs and scales your containers without having to manage servers or clusters. Learn more

Scalable, Durable, Secure Backup & Restore with Amazon S3
AWS Fargate runs and scales your containers without having to manage servers or clusters. Learn more

AWS Marketplace
Find, buy, and deploy popular software products that run on AWS.
Learn more

Have Feedback?:
submit feedback to tell us about your experience with the AWS Management Console.

Feedback English (US) ▾ © 2021, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Figure 1: The AWS Management Console is the central point of contact for managing the cloud provider's services.

For technical support and other resources that help you plan, deploy, and improve your AWS environment, choose a support plan that best fits your use case. To create a support request:

- Log in to the AWS Management Console.
- Select the *Support* drop-down in the upper right corner (Figure 1C) and then *AWS Support Center*.
- Choose *Create case* and select one of the options *Account and billing support*, *Service limit increase*, or *Technical support*.
- Follow the prompts to describe your case: for example, with the error message received, troubleshooting steps you followed, access to the service from the AWS Management Console, AWS command-line interface (AWS CLI), or application programming interface (API) operations.
- Click *Submit*.

Your request ID and the overview are now displayed.

Finding and Using Services

Several methods can help you find the services you need. Simply select the *Services* drop-down (Figure 2A) in the top-level navigation bar to see the list of all services grouped by category (Figure 2B). Alternatively, you can search for particular services, features, Marketplace products, and AWS documentation without leaving the console. The keyboard shortcut Alt + S (Opt + S) jumps directly to the search bar, the right arrow key autocompletes, and the Enter key navigates to the top result. For example, to navigate to the Elastic Cloud Compute (EC2) console, type *ec2* and press Enter.

On the home page, and in the expanded Services menu, you will find two sections: *Favorites* (Figure 2C) and *Recently visited services* (Figure 2D). When you first begin, you will not initially see any entries. For ease of access, you will want to save the services you use most frequently in the list of favorites by selecting *Services* in the navigation bar. In the *Recently visited services* or *All Services* list, go to the name of the service you want to add as a favorite. Click on the star to the left of the service name (Figure 2E).

Learning More About Services

The Management Console provides a variety of learning resources, including articles on use cases, documentation, Getting Started tutorials, on-demand webinars, and deployment templates. All of these resources are available on the Management Console home page and the home pages for each service. For a comprehensive overview of a specific service – from general to specific – visit the following four web pages that each AWS service offers as soon as it is released:

- Overview page: Go to the Products page [3] and select a service to determine whether it fits your use case.
- Features: All the main features are in the second tab of the Services page with detailed descriptions that you can refer to if you have questions about the technical requirements.
- Frequently asked questions: This page is where you can find in-depth details on all facets of a service in a compact question-and-answer format.
- Documentation: When you are ready to test the service directly in the management console, go to the *Resources* item, where you will find links to user guides, developer guides, API references, tutorials, and more.

The Management Console provides a set of simplified and automated workflows and wizards that make testing and deploying an AWS service easier. After accessing a service's home page, an orange button guides you through the startup wizard. More than 100 services offer a free quota for test purposes [4]. After testing,

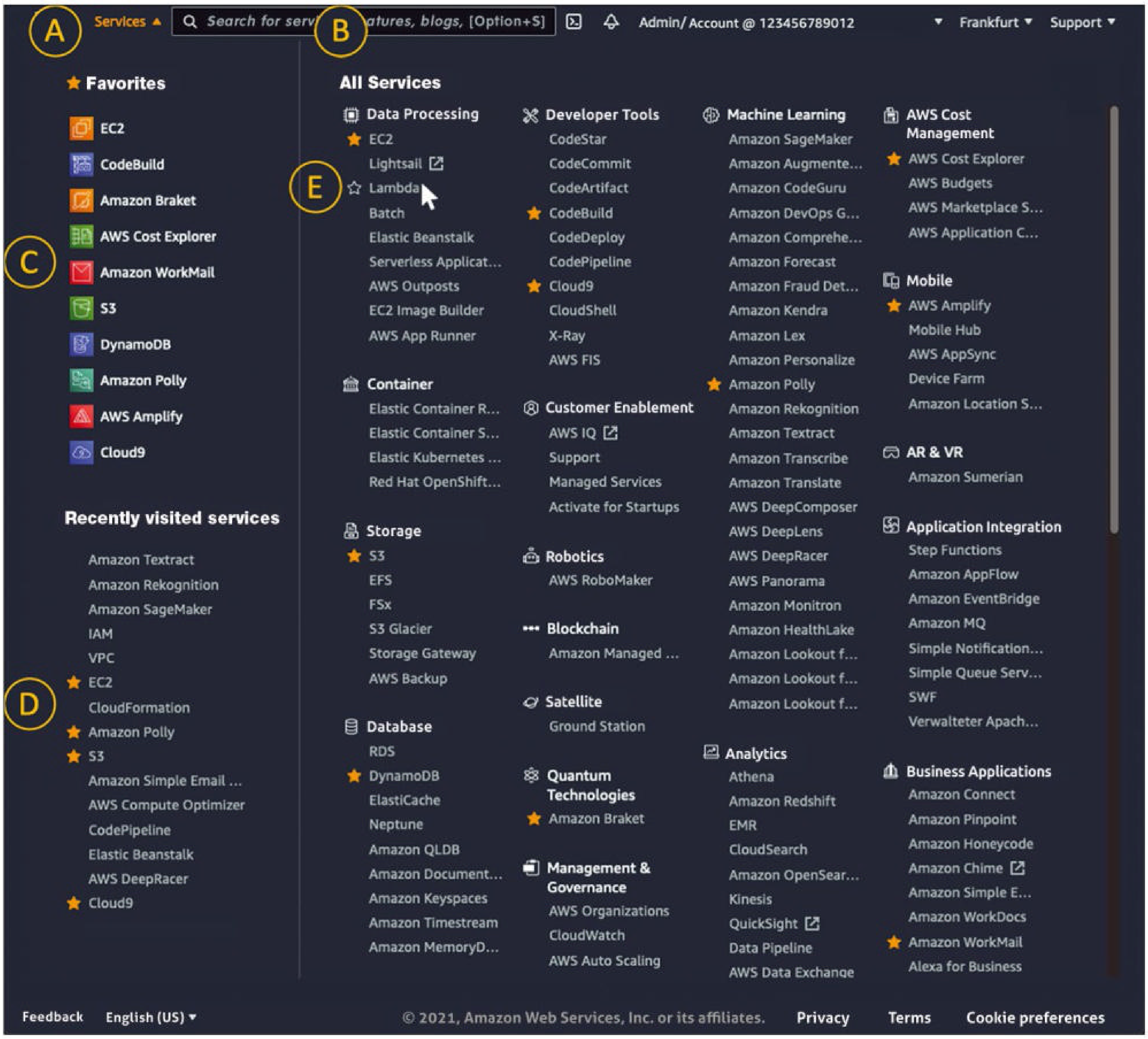


Figure 2: The list of AWS services is extensive but can be narrowed down with the search function to return the services of interest.

remember to stop and delete the resources you created to avoid incurring unintended costs. Also from the Management Console, you can access the AWS Marketplace, a digital catalog with thousands of software offerings from independent providers.

Monitoring Costs, Roles, and Resources

Three basic services – Billing, AWS IAM, and AWS CloudWatch – help establish cost transparency, security, and reliability when you use the AWS cloud. You will therefore definitely want to add the services to your favorites list in the Management Console.

To monitor your service usage, plan expenditures, settle the AWS invoice, and update your billing data, use the Billing and Cost Management dashboard. For example, to set up an alert to receive a notification when expenditures exceed a certain amount that you specify:

- Log in to the AWS Management Console and, in the top right corner of the console, select your account name or number and *My Billing Dashboard*.
- In the navigation menu on the left, go to *Budgets*, click the orange *Create budget* button, and confirm *Cost budget* as the preselected budget type. Set the budget amount that suits you and assign a budget name (e.g., *MyAWSBudget*).
- Create alerts for the following three thresholds and provide your email address to receive notifications for each: 50 percent of forecasted costs, 100 percent of forecasted costs, 100 percent of actual costs.
- Finally, click *Create*; you can skip the *Attach actions* step.

AWS IAM lets you create and manage AWS users and groups, and you can set permissions to allow or deny users access to AWS resources. AWS CloudWatch supplies monitoring and operational data in the form of logs, metrics, and events that offer a unified view of your resources, applications, and services running on AWS and on-premises servers.

Keyboard Aficionados

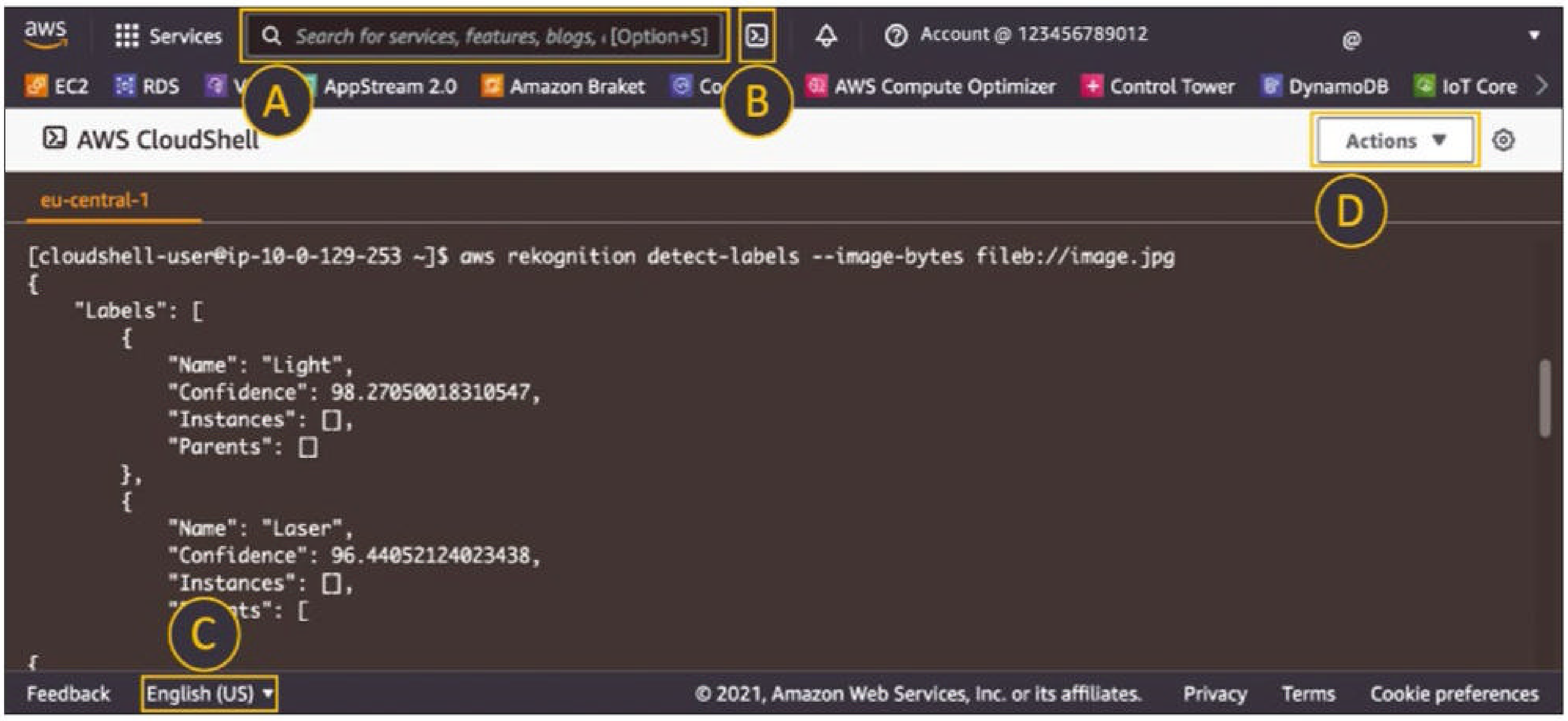
Up to now, the AWS Management Console has been the focus, but it is not always the fastest way to manage the Amazon cloud, especially if you are dealing with multiple accounts, regions, and roles. Therefore, in this second section, I look at command-line tools, SDKs, toolkits for integrated development environments, and automation through Infrastructure as Code.

CLIs let you to manage and script AWS services from the terminal. The CLI is directly available in the Management Console in the form of AWS CloudShell – or easily installed in any other environment. This browser-based command-line environment helps you securely manage, explore, and interact with your AWS resources.

The CLI is pre-authenticated with your permissions from the Management Console, and popular development and operational tools are already installed, removing the need for local installation or configuration. CloudShell lets you run scripts from the AWS CLI, experiment with AWS service APIs with the help of AWS SDKs, or use a variety of other tools to work more efficiently.

CloudShell

In the AWS Management Console, launch CloudShell by typing *cloudshell* in the search box and then selecting the *CloudShell* option (Figure 3A). Next, select the CloudShell icon at top right in the menubar (small box with > symbol; Figure 3B). To select an AWS region, go to the bottom panel and choose a supported AWS region (Figure 3C); the available regions are highlighted. Now select a preinstalled shell by entering it at the command prompt: `bash` (the default), `pwsh`, or `zsh`. The *Actions* item in the CloudShell menu (Figure 3D) lets you change the layout, upload and



```
[cloudshell-user@ip-10-0-129-253 ~]$ aws rekognition detect-labels --image-bytes fileb://image.jpg
{
    "Labels": [
        {
            "Name": "Light",
            "Confidence": 98.27050018310547,
            "Instances": [],
            "Parents": []
        },
        {
            "Name": "Laser",
            "Confidence": 96.44052124023438,
            "Instances": [],
            "Parents": [

{
```

Figure 3: AWS CloudShell is easily customized to suit your needs.

download files, and restart or reset the CloudShell.

Installing the AWS CLI

The AWS CLI is a unified tool for managing your AWS services. With just one piece of software that you can download and configure, you can control multiple AWS services at the command line and automate them with scripts. To interact with AWS through the CLI outside of CloudShell (e.g., on your laptop), you need to configure the credentials for API calls. I will also look at how you can set up multiple profiles to access more than one AWS account, either with additional credentials or by IAM role changes.

Depending on your operating system or preference for using containers, you have several ways to install the CLI. Always follow the latest instructions for installing version 2 of the AWS CLI [5]. After the install, type

```
$ aws -version
  aws-cli/2.2.17 ➘
  Python/3.9.6 ➘
  Darwin/20.5.0 ➘
  source/x86_64 prompt/off
```

The version numbers here are just by way of an example, of course.

The CLI is now installed. You can try to run a command, for example:

```
aws ec2 describe-vpcs
```

However, because your login information is still missing, an error message will pop up.

Configuring Your Credentials

To configure your credentials, you need the

```
aws configure
```

command. You are prompted for the AWS Access Key ID, the AWS Secret Access Key, the default region name, and the default output format.

You can use the credentials of an existing user in AWS IAM or create a new one.

To override the default region, you stipulate the `--region` option with each command (e.g., `--region eu-central-1`). For a list of region codes, see the Regions menu in the AWS Management Console. Finally, the format type specifies how the output will be displayed by default; the options include JavaScript Object Notation (JSON), YAML, and TEXT.

After entering all the data, you should see output like this in the terminal:

```
AWS Access Key ID [None]: ABCDEFGHIJKLM
AWS Secret Access Key [None]: ABCDEFGHIJKLM
Default region name [None]: eu-central-1
Default output format [None]: json
```

Now run the command mentioned earlier,

```
aws ec2 describe-vpcs
```

and you now see output because each new AWS account comes with a default network – Amazon Virtual Private Cloud (VPC) – preconfigured,

```
{
   "vpcs": [
     {
       "CidrBlock": "10.0.0/16",
       "DhcpOptionsId": "dopt-d12345",
       }, "state": "available",
       "VpcId": "vpc-0123456789abc def",
       }, "ownerId": "123456789012",
       ....
```

which confirms that your AWS CLI is set up correctly. Two new files (`config` and `credentials`) have been created, either in `~/.aws` (Linux/macOS) or in `%UserProfile%\.aws` (Windows). The `credentials` file contains the credentials you specified.

Managing S3 with the Python SDK

APIs let you completely manage AWS services. The cloud provider offers SDKs to reduce programming complexity. A wide range of programming languages and frameworks are supported:

- In general: C++, JavaScript, Python, PHP, .NET, Rust, Swift, Kotlin, Ruby, Java, Go, and Node.js
- Hardware-related programming languages: Embedded C and Arduino
- Web: React, Angular, Vue, and Next.js
- Mobile: Android, iOS, React Native, Ionic, and Flutter

For example, the AWS SDK for Python, Boto3 – named after the Amazon River freshwater dolphin – requires at least Python 3.6. Later, I will go through the steps to list all the Amazon Simple Storage Service (S3) buckets (containers for data stored in Amazon S3).

To begin, install the latest Boto3 version by typing:

```
pip install boto3
```

Make sure you have valid AWS credentials. To do so, configure your Access Key and Secret Access Key with `aws configure`, as described earlier.

After you import Boto3 and specify which service you want to use, you are ready to use it. In this example, I work with the Python interactive shell and Amazon S3 object storage:

```
import boto3
s3 = boto3.resource('s3')
```

You now have an S3 resource, so you can send requests to the service. The code

```
for bucket in s3.buckets.all():
    print(bucket.name)
```

uses the S3 bucket API to output all the bucket names.

Toolkit for JetBrains

The majority of developers and administrators use an IDE to help uniformly develop, test, debug, and deploy software and infrastructure. Existing IDEs can be extended with AWS toolkits to interact smoothly with the AWS cloud. Supported IDEs include:

- Eclipse
- AWS Cloud9
- Microsoft: Visual Studio, Visual Studio Code, and Azure DevOps

- JetBrains: CLion, GoLand, IntelliJ, WebStorm, Rider, PhpStorm, PyCharm, RubyMine, and DataGrip

By way of an example, I look at JetBrains. As a prerequisite, you first need to install and launch a supported JetBrains IDE. Next, open the Preferences, select *Plugins* and click the *Marketplace* tab. In the search box, enter *AWS Toolkit*. When *AWS Toolkit by Amazon Web Services* appears, select it, install the plugin, and restart the IDE. To connect to an AWS account for the first time, select *AWS: No credentials selected* in the status bar (Figure 4).

Now, either choose an AWS profile configured in the previous two sections or edit the AWS credentials in *Edit AWS Credential file(s)*. To test the configuration, select the *CloudWatch Logs* node to see the list of log groups. Double-click on the name of a log group and select a log stream to read the specific log messages.

Automation Through IaC

You can use Infrastructure as Code (IaC) to model and set up your AWS resources, so you spend less time managing them and more time focusing on the applications you want to run in AWS. To begin, create a template that describes all of the AWS resources you want, such as EC2 instances or Amazon S3 buckets.

The CloudFormation service takes care of provisioning and configuring these resources for you: You don't have to create, configure, and figure out individually what depends on what in the AWS Management Console user interface – CloudFormation does that work for you.

A CloudFormation template describes your desired resources and their dependencies, so you can launch and configure them together as a stack. With a template, you can create, update, and delete an entire stack as a single entity, rather than managing resources individually. Stacks can be managed and deployed across multiple AWS accounts and regions. The template is saved as a text file, and its format complies with the JSON or YAML standard. Because they are text files, you create and edit them in any text editor and manage them in your version control system along with the rest of the source code.

The template describes the AWS resources you want to create and configure. The following basic template describes a single resource of the `AWS::S3::Bucket` type:

```
{"
Resources" : {"
HelloBucket" : {
"Type" : "AWS::S3::Bucket"
}}
```

Note that you need to choose a unique name for your S3 bucket (e.g., `HelloBucket-<123-full-number>`) that no other AWS customer has already chosen. When you use this template to create a stack, CloudFormation easily creates an S3 bucket because the service can simply use the default settings. For other resources, such as an EC2 autoscaling or EC2 instance, CloudFormation needs more information.

IaC with Programming Languages

Instead of writing the CloudFormation templates yourself, you can use a programming language to define your resources. The AWS Cloud Development Kit (AWS CDK), for example, supports well-known programming languages and leverages their power to express complex concepts. Cloud resources are preconfigured with tried and trusted defaults and then rolled out in a secure, repeatable manner as CloudFormation templates. The earlier short example with CDK in TypeScript would be:

```
const helloBucket = ↲
  new s3.Bucket(this, 'HelloBucket');
```

JavaScript, Python, Java, and C# are also possible. For more information, see the CDK website [6] or the CDK workshop [7].

Conclusions

Amazon offers numerous services in the cloud in the form of AWS. In the first part of the article, I looked at the essential features of the AWS Management Console. In the second part, I focused on how you can completely replace point and click in the AWS Management Console with command-line tools, SDKs, other tools, and IaC. Although I could only provide a brief insight into each category, the options for managing the AWS cloud without the web interface are fast, repeatable, and targeted, whether you opt for the terminal, scripting, or coding in your programming language of choice. ■

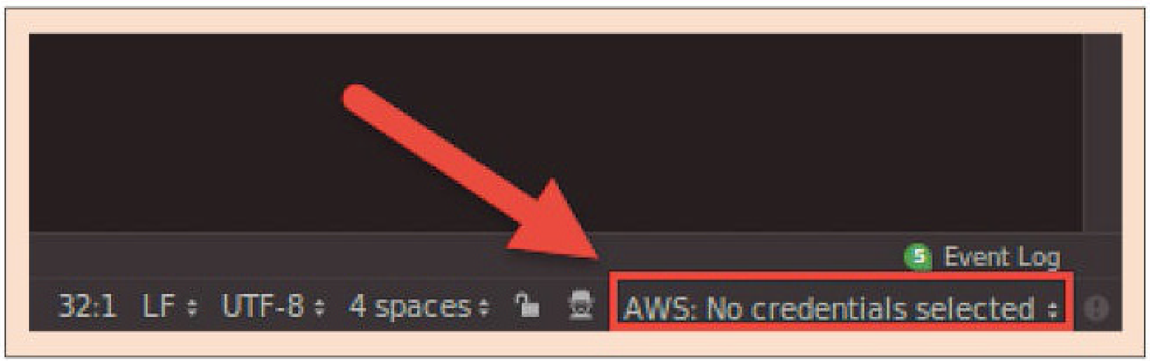


Figure 4: Before using the AWS toolkit for JetBrains for the first time, a click on the status bar is required to connect to an AWS account.

Info

[1] AWS sign-in: [console.aws.amazon.com]

[2] Global infrastructure in AWS: [https://aws.amazon.com/about-aws/global-infrastructure/]

[3] Products: [https://aws.amazon.com/products/]

[4] AWS Free Tier: [https://aws.amazon.com/free/]

[5] Configuring AWS CLI: [https://docs.aws.amazon.com/en_us/cli/latest/userguide/cli-chap-configure.html]

[6] AWS CDK: [https://aws.amazon.com/cdk/]

[7] AWS CDK Intro Workshop: [https://cdkworkshop.com]

Author

Christopher Henkel is a Solutions Architect at AWS.

Photo by Alessandro Erbetta on Unsplash

Azure AD and AD Domain Services for SMEs

Above the Clouds

Azure Active Directory Domain Services is a Microsoft product, distinct from Active Directory and Azure Active Directory, that offers centralized directory services in the cloud in place of an often convoluted on-premises operation. By Martin Loschwitz

Small- and mid-sized enterprises (SMEs) have some of the same IT needs as larger enterprises. Unlike large companies, however, SMEs usually do not have a separate department to look after internal IT. This work is either outsourced to external service providers or an employee performs the task on the side.

To deal with various compliance issues and to simplify the management of user access and devices, central directory services such as the Lightweight Directory Access Protocol (LDAP) or Active Directory have become increasingly widespread. Although a full-blown Active Directory was once considered a tool needed only by huge company infrastructures, small Active Directory setups can now also be found in companies with around 10 or more employees. The problem is that an Active Directory setup is hardly less complex today than it was a few years ago; if anything, it has become even more difficult.

As a central directory service, Active Directory also has a central relevance. If all user logins rely on Active Directory, they will not work if it is not available. Therefore, administrators see themselves confronted with the issue of needing high availability, even though it is often impossible to achieve on site because typically no data center infrastructure is available and the server with Active Directory might be hidden away in a broom closet.

Microsoft's focus on the cloud in the form of Azure could provide a solution. Lo and behold, if you scroll through the product portfolio, you will find Azure Active Directory (Azure AD) [1]. However, because of how it is structured, Azure AD is not the classic directory service that you know from your local Active Directory, even if the name suggests a relationship with its namesake from the on-premises world. The difference between the two is easy to see because Microsoft exclusively offers Azure AD as a managed service (i.e., as a Platform-as-a-Service (PaaS) or Software-as-a-Service offering (SaaS)). The user does not even find out where the service is running or what the underlying system looks like. The primary task of Azure AD is to store combinations of user names and passwords and make them available to other services and applications over defined standard interfaces (e.g., in the context of OAuth). Indeed, the functional scope of Azure AD is more similar to the idea of a central vault for user data with a cloud connection than to that of a central directory service.

The good news is that Microsoft now also offers a "real" Active Directory in its cloud – Azure Active Directory Domain Services (Azure AD DS, sometimes called AAD DS) [2] – although it is far less in the spotlight than Azure AD. In this article I look at how admins in smaller companies can benefit from Microsoft's directory services in the cloud, how Azure AD and Azure AD DS differ, whether Server Message Block (SMB) IT managers can get rid of the annoying issue of Active Directory operations by migrating the service to the cloud, and which service is the best fit for various use cases.

Away with Old Habits

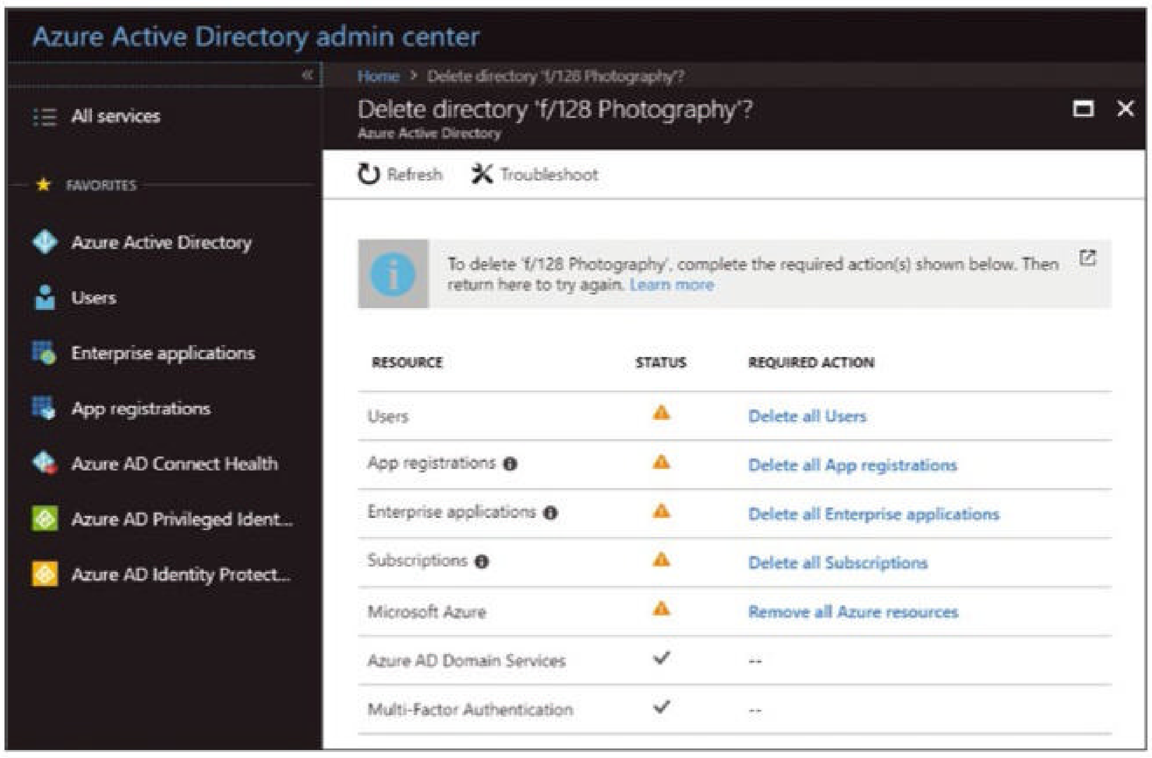
The Azure AD design is not a coincidence or the result of incompetence by those responsible for it at Microsoft. Azure AD was designed from the start to be exactly the service it is today. And Microsoft took the opportunity to drop off some of the legacy ballast that Active Directory is still dragging along with it today, as a look at the protocols used quickly shows. Although Active Directory still uses some of the same protocols it did several decades ago, Azure AD is primarily accessed through API interfaces according to the REST standard.

On the one hand, this design makes it far easier for Microsoft to connect Azure AD to other cloud services and applications. The company makes it abundantly clear how this works with its own cloud applications: Microsoft 365 (formerly Office 365), for example, can be operated fully with users from Azure AD. On the other hand, this design also means that Azure AD is unsuitable as a replacement for the local Active Directory because it lacks a great deal of functionality (Figure 1). Computers cannot be integrated into a domain in Azure AD, not least because Azure AD does not implement the domain concept. If you want to assign rights by group policies, you are out of luck because they do not exist in Azure AD – nor do LDAP, NT LAN Manager (NTLM), or Kerberos. Moreover, the hierarchy provided in Active Directory, which enables organizational units (OUs), is also completely missing in Azure AD.

What does work with Azure AD, however, is the single-source-of-truth principle when it comes to user access, because Azure AD provides a way to synchronize users and passwords from existing Active Directory setups. When companies use software that can attach to Azure AD as an authentication and authorization service, synchronizing the on-premises Active Directory with Azure AD allows the same user base to be used everywhere. Microsoft even offers a way out for applications that cannot talk to Azure AD directly: The Azure Application Proxy can be deployed upstream of conventional web applications but communicate with Azure AD in the background. If the user name and password match, the application proxy forwards the client to a secure website (e.g., by HTTP authentication) and receives the correct combination of user name and password from the application proxy.

Almost Like a Local AD

A quick look at the Azure AD DS feature list gives Microsoft administrators that cozy feeling of being at home far more than is the case with Azure AD. In practical terms, Azure AD DS is a domain controller redundantly operated by Microsoft in the Azure cloud; it handles most of the familiar functionality of local setups in the usual way. Joining your notebooks and servers to the remote Active Directory domain is no problem. Azure AD DS also can be used to give the applications in your IT environment their user data over LDAP, either by public IP address over the Internet or a virtual private network (VPN) to the Azure cloud, which Microsoft also offers. Azure AD DS supports Kerberos authentication and the NTLM protocol, as well as nested OUs and group policies. Azure AD DS is therefore far closer to a conventional, on-premises Active Directory than Azure AD.


Azure Active Directory admin center
All services
FAVORITES
Azure Active Directory
Users
Enterprise applications
App registrations
Azure AD Connect Health
Azure AD Privileged Ident...
Azure AD Identity Protect...
Home > Delete directory 'f/128 Photography'?
Refresh
Troubleshoot

Delete directory 'f/128 Photography'?
Azure Active Directory

To delete 'f/128 Photography', complete the required action(s) shown below. Then return here to try again. Learn more

RESOURCE	STATUS	REQUIRED ACTION
Users	⚠	Delete all Users
App registrations ❶	⚠	Delete all App registrations
Enterprise applications ❶	⚠	Delete all Enterprise applications
Subscriptions ❶	⚠	Delete all Subscriptions
Microsoft Azure	⚠	Remove all Azure resources
Azure AD Domain Services	✓	--
Multi-Factor Authentication	✓	--

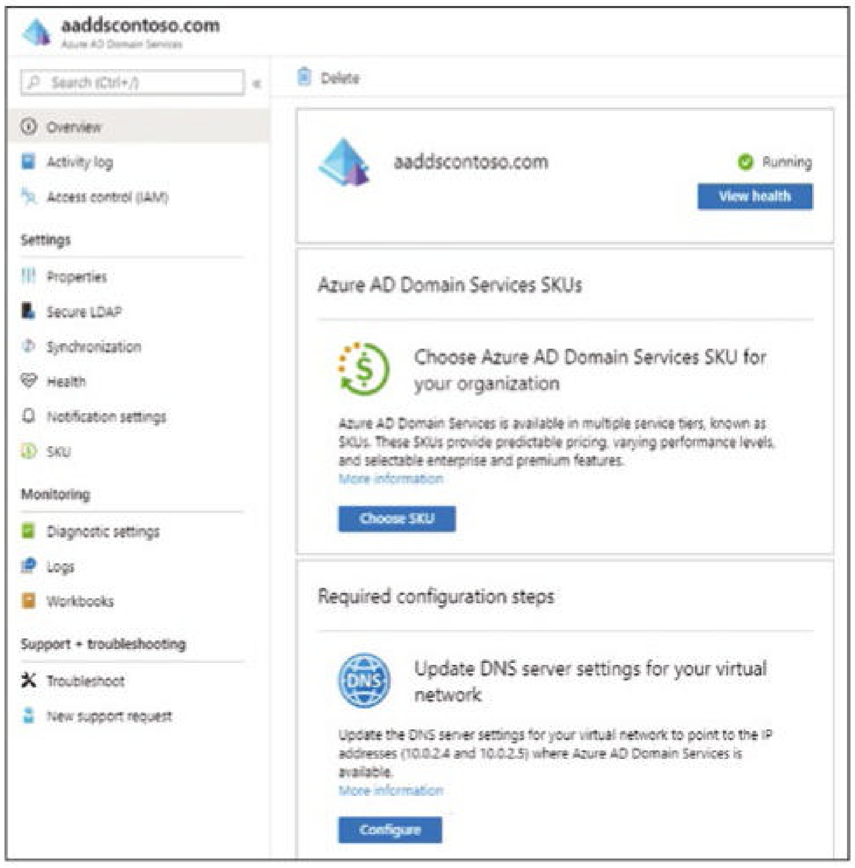
Figure 1: Azure AD is a kind of password vault for the cloud with connectivity to many other services, but it is not a full-fledged replacement for Active Directory.

Even better news, from the administrator's point of view, is that Microsoft operates Azure AD DS instances completely without user intervention as PaaS. Therefore, you do not have to worry about redundancy or about patching the underlying operating system or Active Directory itself. However, there is a catch: The typical permissions for domain administrators are implemented in Azure AD DS, but the service does not forward them to the user. Domain configuration details are therefore only possible with the GUIs provided by Azure for this purpose, not – as expected by well-versed Active Directory admins – by direct access to Active Directory as a domain administrator, although this situation is unlikely to worry most administrators of Active Directory instances in SMEs (Figure 2). In many places, Active Directory is used as a central directory service for managing users and resources, and Azure AD DS, hosted by Microsoft in its own cloud, is more than up to this task.

The Right Migration Path

Once a company has decided to migrate Active Directory to the cloud, the question of "how" is often on the agenda. SMEs do not need to worry – even less so if Active Directory, as is almost always the case in such setups, has been used as the central logon and authorization service. The user access data can be migrated very easily, removing the need for comprehensive migration planning.

Azure AD, an old acquaintance, accompanies the migration. In the first step, you create a client in Azure AD, which can be interconnected and synchronized with an on-premises Active Directory. Then, in Azure AD DS,



Azure AD Domain Services SKUs

Choose Azure AD Domain Services SKU for your organization

Azure AD Domain Services is available in multiple service tiers, known as SKUs. These SKUs provide predictable pricing, varying performance levels, and selectable enterprise and premium features.

Required configuration steps

Update DNS server settings for your virtual network

Update the DNS server settings for your virtual network to point to the IP addresses (10.0.2.4 and 10.0.2.5) where Azure AD Domain Services is available.

Figure 2: Azure Active Directory Domain Services is an almost complete replacement for an on-premises Active Directory in the cloud. The few missing features are unlikely to be relevant for SMEs.

you configure the new domain that replaces the previous on-premises Active Directory domain. Finally, you just set up the synchronization between Azure AD and Azure AD DS. Once this is complete, the user data from what are three directories at this point is identical, and the on-premises Active Directory can be retired. It's a bit more complex if machines also need to be migrated to the cloud with their Active Directory connections. If you are not dealing with hundreds of end devices, the approach of individually unregistering the devices from the old domain and registering them with the new domain is probably your best bet.

Privacy vs. Security

If you live outside the US, you might be a bit nervous at the prospect of migrating services to a public cloud, especially one offered by a US provider. After all, according to the often-cited criticism, companies in the US have to grant the US government full access to their stored data, which is not in compliance with the GDPR, because data within the sense of Article 44 [3] is transferred to a third country that either has not agreed to uphold the GDPR or with which a comparable agreement on data protection does not exist.

In the context of Azure AD DS in particular, however, this criticism is not truly valid for several reasons.

When companies use Azure AD DS, they are only migrating some of their data to the cloud – the usernames, passwords, and any additional metadata for individual devices. Switching from the local Active Directory to an Active Directory in the cloud does not mean immediately uploading the entire content of your local storage to the Internet, as is sometimes the case with other services. Of course, login names and passwords are sensitive, but they do not directly enable access to a company's business data, for example.

Additionally, many SME admins underestimate another factor: Operating an Active Directory "correctly" and securely causes overhead that should not be underestimated. I have already addressed the problem of needing a meaningful high-availability setup without having the infrastructure to build one. Additionally, security updates for services such as Active Directory or the underlying Windows often have to be installed at very short notice, which is sometimes a bumpy ride and means even more work. Because it is difficult to find the right time for an Active Directory restart, which effectively shuts down the office for several minutes, some administrators ignore the issue of security altogether, which in turn opens the door to attackers, meaning you then have to worry about data security at least as much as you would in the cloud.

As a huge provider, Microsoft is used to protecting online services and data. It is less likely that attackers will invade Azure and take data with them than that an attack on your broom closet will be successful and your Active Directory the victim. Finally, Azure AD DS comes with automatic backups, which you can download if required. Even if an attack does succeed, Azure AD DS still offers the option of returning to a secure backup of user data from the past – at least if the start of the attack can be clearly identified.

Conclusions

Azure Active Directory Domain Services can be a real alternative to on-premises Active Directory, especially for SMEs. Many of the tedious maintenance tasks in the Active Directory context are eliminated in the cloud variant; at the same time, SMEs will typically not need any of the functions that are missing in Azure AD DS. In any case, anyone who spends a lot of time maintaining an on-premises Active Directory that would be better invested elsewhere will also want to take a closer look at Azure AD DS. Admittedly, it adds a line item to your operating expense list because Azure AD DS is based on a subscription model like most Azure services, with monthly fees for use. For SMEs in particular, however, the charges should be within tolerable limits and, in any case, not exceed the costs incurred by several hours of work on Active Directory by an average IT service provider. ■

Info

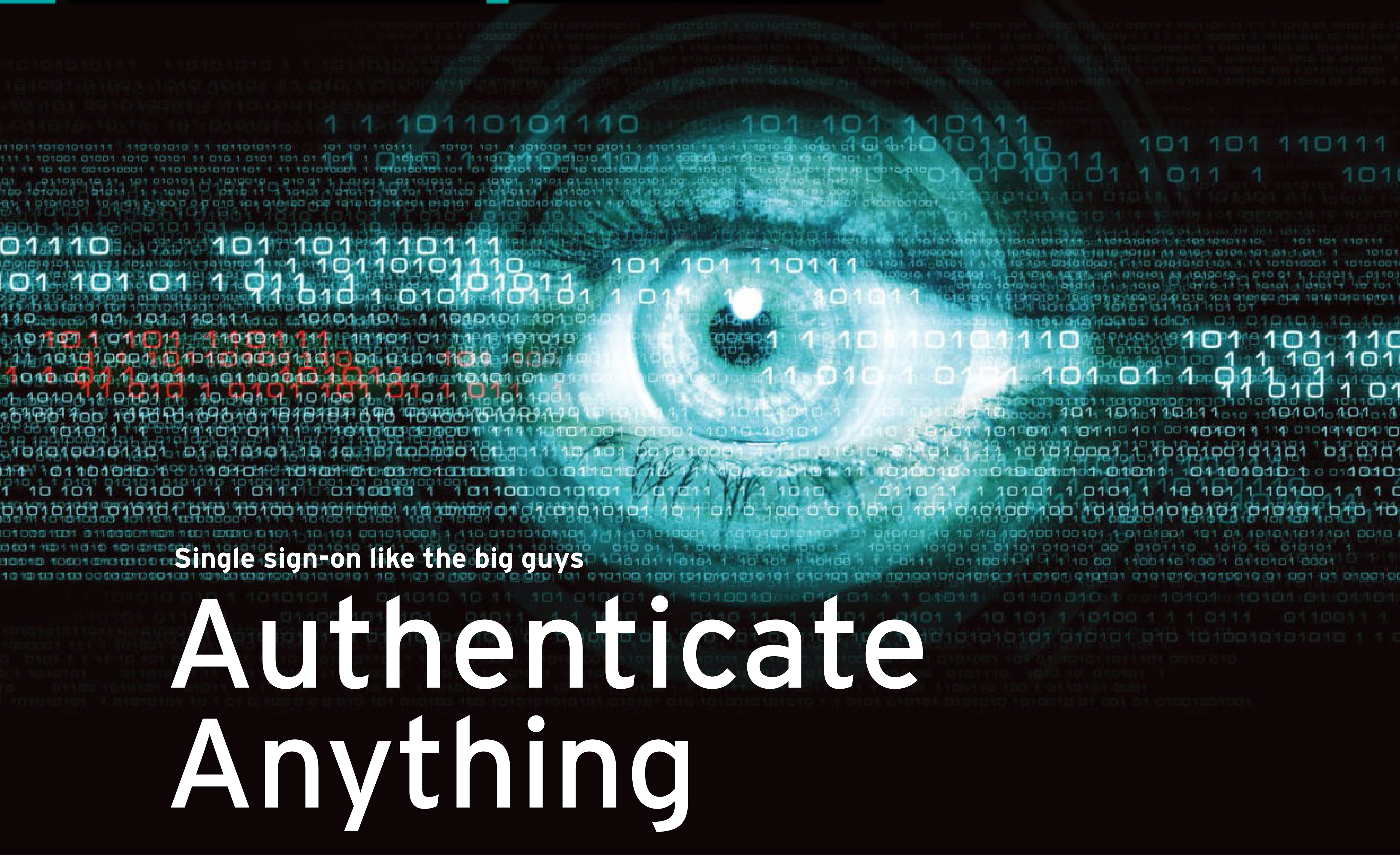
[1] Azure AD: [https://azure.microsoft.com/en-us/services/active-directory/]

[2] Azure AD DS: [https://azure.microsoft.com/en-us/services/active-directory-ds/]

[3] GDPR Article 44: [https://gdpr-info.eu/art-44-gdpr/]

The Author

Freelance journalist Martin Gerhard Loschwitz focuses primarily on topics such as OpenStack, Kubernetes, and Ceph.



Single sign-on like the big guys

Authenticate Anything

Keycloak is a robust and mature project that provides a modern single sign-on authorization experience and centralized authentication of your apps. By Stefano Chittaro

Once your set of internal applications grows greater than an order of 10s, you can end up in a scenario where credentials storage for each service gets out of control. Users might start complaining about how difficult it is to handle multiple passwords, and your day could turn into a password reset ticket nightmare. If you wonder whether a single sign-on (SSO) experience à la Google and Amazon is possible, even at a smaller scale, the answer is "Yes"! Keycloak can do exactly that.

A comprehensive administration introduction to Keycloak appeared previously in *ADMIN* [1], so in this article you will travel through the other end of the spectrum: How to enable your application with proper SSO, with or without writing code.

The Keycloak Project

Keycloak is a mature free and open source software (FOSS) project whose first production release goes back to the year 2014 [2]. It's largely funded and developed by Red Hat, and it is the software on which their SSO commercial offering is based. The tool's goal is to provide a modern and secure SSO experience for any application on the basis of either the OIDC or SAML framework (see the "OIDC vs. SAML" box).

Until version 16, inclusive, Keycloak ran on top of the WildFly application server (formerly JBoss). Since version 17, however, the project has shifted to Quarkus, breaking some configurations but gaining in performance and general lightness.

SSO Benefits

You have surely dealt with at least one user management application, the most famous of which is Microsoft Active Directory (see the "User Brokering" box). Moreover, you might have run into independent applications that run their own user databases. With SSO, you can reduce the complexity of working with these applications. The major benefits are:

- Managing your users all in one place
- Applications won't need to store user data or passwords
- Applications will benefit from password management or two-factor authentication out of the box

Getting Started

To begin, you will deploy a Keycloak instance with the official Docker

OIDC vs. SAML

OpenID Connect (OIDC) is the only authentication framework used in this article, although Security Assertion Markup Language (SAML) is widely used and supported, especially in the Enterprise segment. The choice usually falls on OIDC because of its increasing popularity, lightness, and simplifications like data exchange by JSON instead of XML.

User Brokering

If Active Directory is where you store your user backend, or you want to set up alternative social logins for your users, Keycloak can act as an authentication broker connected to either LDAP/Kerberos or other SAML/OIDC identity providers. If configured in this way, it will check your credentials against a third party before allowing the underlying application to be accessed.

Lead Image © Sergey Nivens, 123RF.com

image (see the “Get Docker-Ready” box). The application will be exposed on port 8080 and will use an ad hoc Postgres instance for its data and configuration storage (Listing 1).

Once the `docker-compose.yml` file is ready, start the service by typing

```
docker-compose up -d
```

After the startup phase is complete, you reach the Keycloak admin interface on *http://localhost:8080*.

Configuration and User Creation

Now you’ll need configurations for:

- A test authentication realm
- A `client` for the native application
- A `client` for the application authenticated through a proxy
- A user that will log in against both clients

It might sound counterintuitive, but in Keycloak-speak a client is an application that authenticates against Keycloak, so always remember that when a *client* is mentioned, it can refer also to a service or server and not just to front-end web apps.

Once you are logged in, mouse over the Master section in the upper left corner and click *Add realm*; enter *test-realm* as a name and submit.

Next, go to the Clients section and click *Add Client*, set the Client ID to *native*, and set the Root URL to *http://localhost:5000* before clicking *Save*. Once the settings are saved, switch the `Access Type` to *confidential* and save again.

Now repeat the last (add client) procedure, but this time set the Client ID to `proxy` and the Root URL to *http://localhost:4180*. Additionally, click on the *Mappers* tab and create a mapper called *audience*, of *Audience* type, and select *proxy* as the included target audience.

Finally, go to the *Users* section, click *Add User*, and fill in the form with anything you like (Figure 1). Set both Enabled and Email Verified to *ON*, then click *Save*. Before leaving this section, click on the *Credentials* tab and set a password for your newly created user.

A Native Authentication App

Keycloak used to provide libraries for several languages called *Adapters*. Since February 2022 they have been discontinued because a robust availability of independent OAuth2 and OIDC libraries has been reached.

As an example, I’ll set up a Python app that uses Flask and the Flask-OIDC module. You’ll need Python 3.8 or later, PIP, and Python-venv. On Ubuntu or Arch Linux type,

```
apt install python3 python3-venv ⮐
  python3-pip
pacman -S python python-pip
```

Listing 1: Deploying Keycloak

```
version: '2.4'
services:
  keycloak:
      container_name: keycloak
      image: quay.io/keycloak/keycloak:17.0.1
      ports:
       - 8080:8080
      environment:
        - KEYCLOAK_ADMIN=admin
        - KEYCLOAK_ADMIN_PASSWORD=SOME_PASSWORD
        - KC_DB=postgres
        - KC_DB_URL=jdbc:postgresql://postgres:5432/
                                        keycloak
        - KC_DB_USERNAME=postgres
        - KC_DB_PASSWORD=SOME_DB_PASSWORD
      command: ["start-dev"]

  postgres:
      container_name: postgres
      image: postgres:14
      environment:
          - POSTGRES_PASSWORD=SOME_DB_PASSWORD
          - POSTGRES_DB=keycloak
          - PGDATA=/var/lib/postgresql/data/pgdata
      volumes:
          - pgdata:/var/lib/postgresql/data/pgdata

volumes:
  pgdata:
```

respectively. Now create a file named `app.py` with the content of Listing 2 and another called `oidc-config.json` with the content of Listing 3.

Note that for the authentication flow to work, you must replace line 7 of

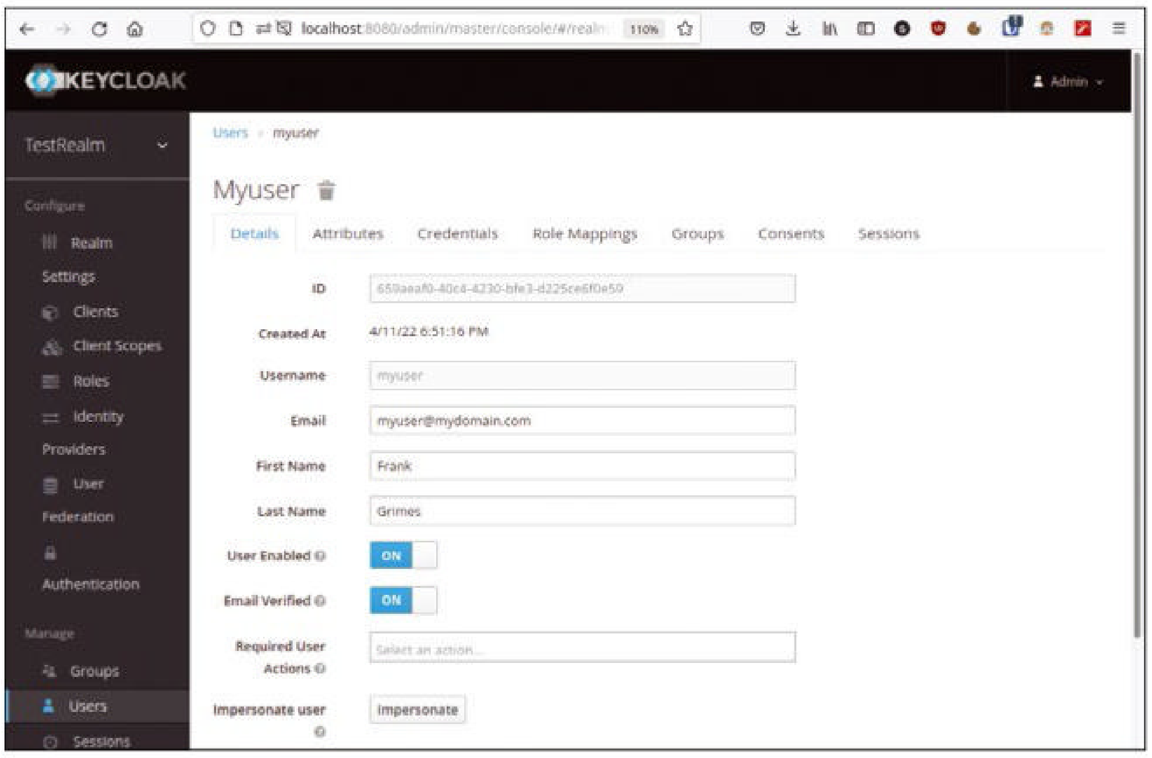
Get Docker-Ready

Throughout the article, I make use of Docker to spin up services quickly and without installing unnecessary packages. To do so, the Docker engine installation is required, which can be accomplished with the one-liner:

```
# curl -sSL https://get.docker.com | ⮐
  sudo bash -
```

This command fetches the latest official installation script, detects which Linux distribution you’re running, adds the proper package manager repositories, and installs the engine.

Make sure, though, as a conscientious admin, to check the content of a script every time you plan to pipe it directly to `sudo bash`.



ID: 650aaaf0-40c4-4230-bfe3-d225ce6f0e50

Created At: 4/11/22 6:51:16 PM

Username: myuser

Email: myuser@mydomain.com

First Name: Frank

Last Name: Grimes

User Enabled: ON

Email Verified: ON

Required User Actions: Select an action...

Impersonate user: Impersonate

Figure 1: User details on the Keycloak admin user interface.

Listing 2 and line 4 of Listing 3 with the client key found in the *Credentials* tab of the Keycloak *native* client created for this example.

To create the environment to run the application, enter

```
python -m venv .venv && ↩
  source .venv/bin/activate
pip install flask flask-oidc ↩
  itsdangerous==2.0.1
```

Listing 2: app.py (Native)

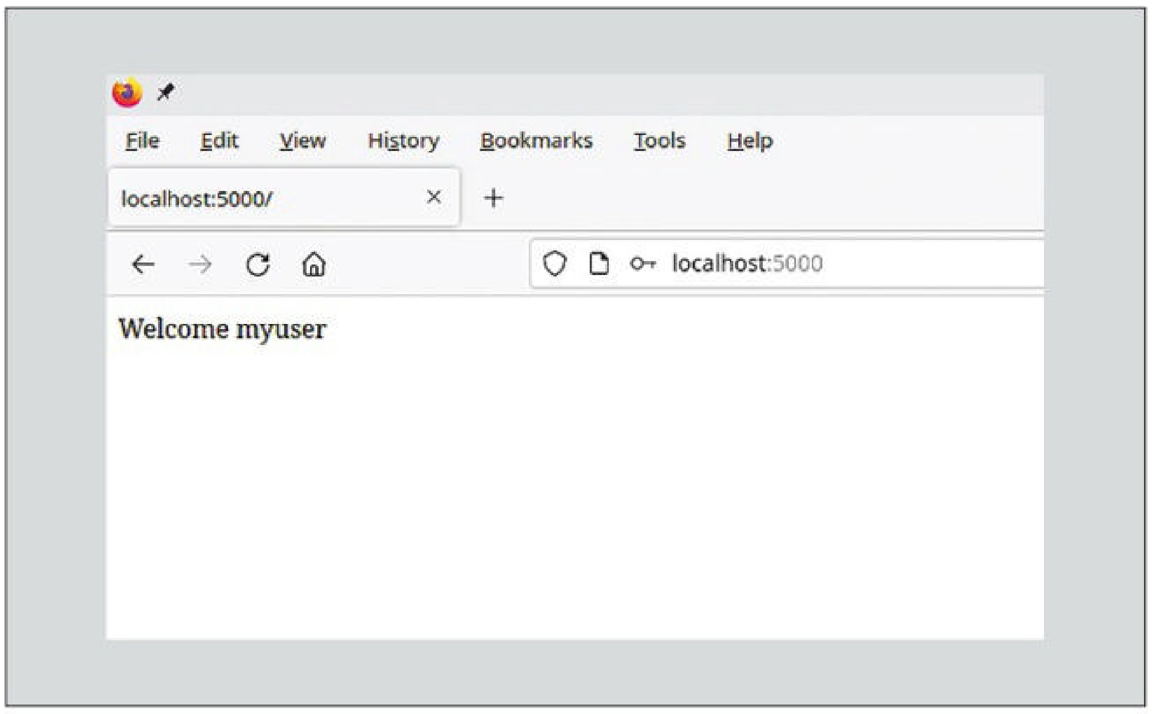
```
01 # Import dependencies
02 from flask import Flask, g
03 from flask_oidc import OpenIDConnect
04
05 # Create a Flask app and set the OpenID connect config params
06 app = Flask(__name__)
07 app.secret_key = 'hu5t82LXfYKtr3XFcPeAYaCBWFK8DcI1'
08 app.config['OIDC_CLIENT_SECRETS'] = 'oidc-config.json'
09 app.config['OIDC_COOKIE_SECURE'] = False
10
11 # Instantiate a flask_oidc object that will be used to handle the authentication flow
12 oidc = OpenIDConnect(app)
13
14 # Function decorators to set the app route and the required login
15 @app.route('/')
16 @oidc.require_login
17 def index():
18    # Check if user is authenticated
19    if oidc.user_loggedin:
20        # If user is authenticated, run this code
21        return 'Welcome %s' % oidc.user_getfield('preferred_username')
22    else:
23        # If he's not, run this code
24        return 'Not logged in'
```

Listing 3: oidc-config.json

```
01 {
02   "web": {
03     "client_id": "native",
04     "client_secret": "hu5t82LXfYKtr3XFcPeAYaCBWFK8DcI1",
05     "auth_uri": "http://localhost:8080/realms/testrealm/protocol/openid-connect/auth",
06     "token_uri": "http://localhost:8080/realms/testrealm/protocol/openid-connect/token",
07     "issuer": "http://localhost:8080/realms/testrealm",
08     "userinfo_uri": "http://localhost:8080/realms/testrealm/protocol/openid-connect/userinfo",
09     "redirect_uris": [
10       "http://localhost:5000/oidc/callback"
11     ]
12   }
13 }
```

and run the application with the `flask run` command before heading to *http://localhost:5000*.

If everything was done correctly, you should be redirected automatically to the Keycloak instance, where you are asked to authenticate. If the authentication is successful, you will be brought back to the sample application, where the username is printed (Figure 2).



Welcome myuser

Figure 2: After authentication, the app returns your username.

As you may have guessed, the application will have access to a wide range of information on the authenticated user. Such information can be used to template your app or apply further conditions according to the user roles (authorization).

Authentication by Proxy

Sometimes you cannot customize the application for the described mechanism to be implemented, as could certainly be the case for proprietary applications or scenarios where messing with the code is simply not possible. In this case, you can use a reverse proxy that performs the authentication flow on behalf of the application it's serving, forwarding only requests from logged-in users (Figure 3).

One example is OAuth2 Proxy [3], a mature project that aims to provide a layer of authentication in front of an existing application. It doesn't necessarily features all the bells and whistles of more renowned solutions such as Nginx, but it can be used in conjunction with it and is natively cloud friendly, providing Docker images and Kubernetes operators.

To test OAuth2 Proxy, I'll first write

a trivial web app (Listing 4) and, as before, run it with

```
flask run -p 9190
```

The app is now freely accessible at *http://localhost:9190*.

To add the authentication layer, you will need to create an OAuth2 Proxy config file (Listing 5) modifying the `client_id` and `client_secret` with the information from the Keycloak *proxy* client created earlier. The reason for setting the *audience* mapper is that OAuth2 Proxy expects said OIDC field valued with the same name of the configured `client_id`.

Now spin up an instance of the service through a convenient Docker container:

```
docker run --network host ↵
  -v $(pwd)/oauth2proxy.conf:↵
     /etc/oauth2-proxy.cfg ↵
  bitnami/oauth2-proxy:7.2.1 ↵
  --config=/etc/oauth2-proxy.cfg
```

Point your browser at *http://localhost:4180*. If everything went according to the plan, you should be asked by OAuth2 Proxy to authenticate (Figure 4). After a successful procedure, you should be brought back to the trivial "Hello World" app.

Conclusions

With little effort, you can set up your local SSO environment. Of course, further steps are needed to set up a production-ready environment [4], but the goal of this exercise was just to get started. With the use of FLOSS components only, you can achieve a consolidated SSO experience already familiar to your users, saving you time and tickets.

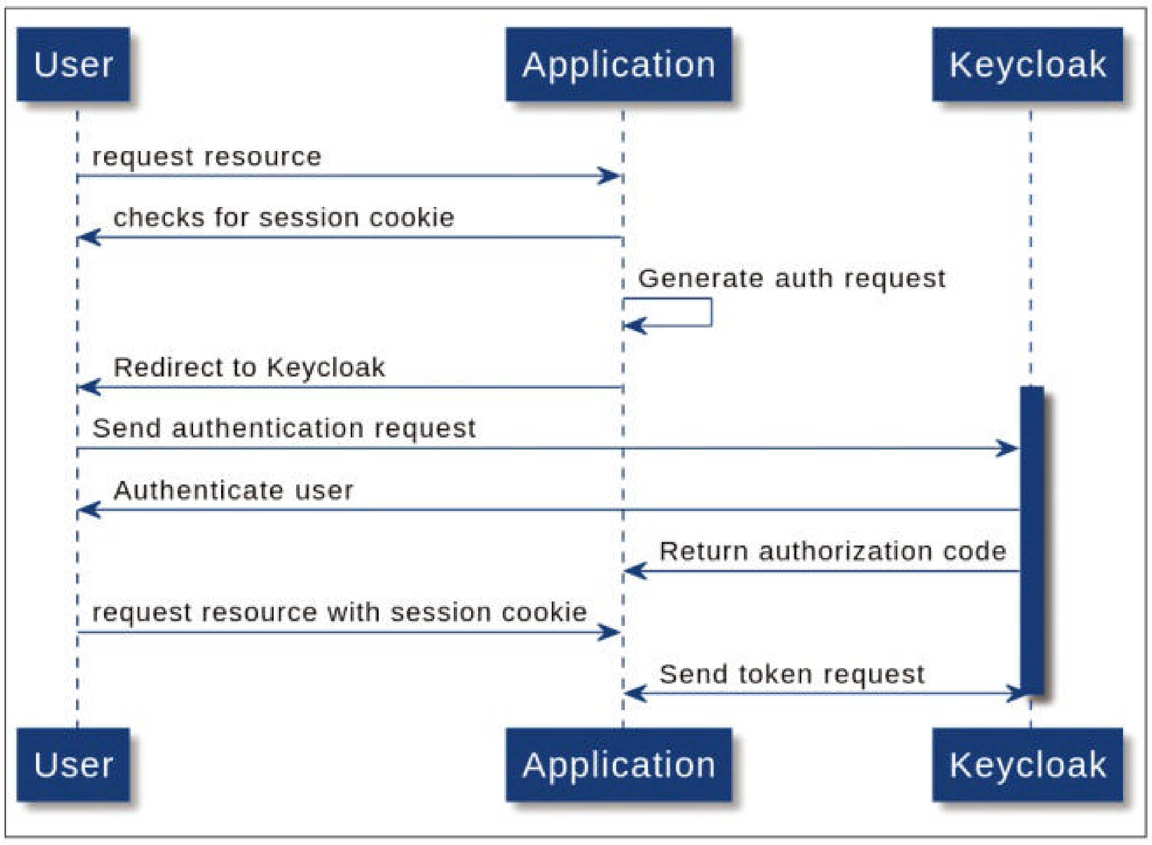


Figure 3: The authentication flow.

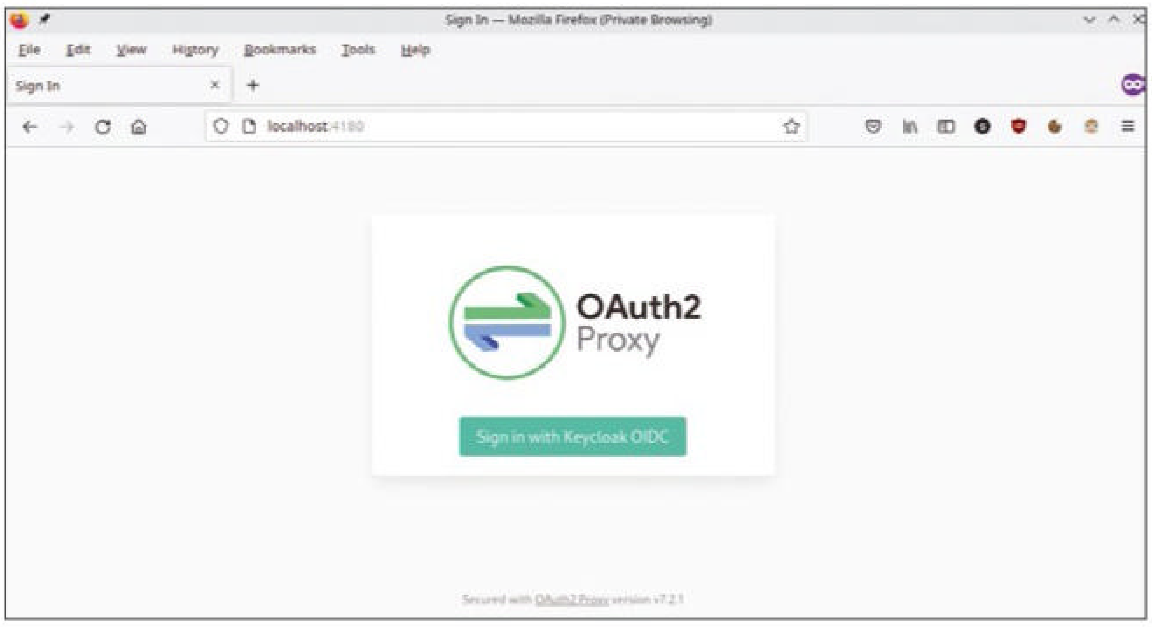

Sign In — Mozilla Firefox (Private Browsing)
File Edit View History Bookmarks Tools Help
Sign In
localhost:4180
OAuth2 Proxy
Sign in with Keycloak OIDC
Secured with OAuth2 Proxy version v7.2.1

Figure 4: OAuth2 Proxy starting the authentication flow.

Info

[1] "Single sign-on with Keycloak" by Matthias Wübbeling, 2021, issue 61, pg. 64, [https://www.admin-magazine.com/Archive/2021/61/Single-sign-on-with-Keycloak/]

[2] Keycloak documentation: [https://www.keycloak.org/guides]

[3] OAuth2 Proxy project: [https://oauth2-proxy.github.io/oauth2-proxy/]

[4] Securing Keycloak for production: [https://www.keycloak.org/server/configuration-production]

Author

Stefano Chittaro manages multicloud deployments with a special focus on automation and observability. Sometimes he rants about technology on [https://nevarsin.blog].

Listing 4: app.py (Proxied)

```
from flask import Flask, g
app = Flask(__name__)
@app.route('/')
def hello():
   return("Hello world")
```

Listing 5: oauth2proxy.cfg

```
# Ports to listen to
http_address = ":4180"
redirect_url = "http://localhost:4180/oauth2/callback"

# The http url(s) of the upstream endpoint.
upstreams = [
   "http://localhost:9190"
]
# Accept requests from all user email domains
email_domains = "*"
# Keycloak configuration
oidc_issuer_url = "http://localhost:8080/realms/testrealm"
provider = "keycloak-oidc"
client_id = "proxy"
client_secret = "t1BmFL4HHk7xC6nPYJEMfiprMfw1QK7b"
# Cookie settings
cookie_name = "_oauth2_proxy"
cookie_secret = "4235342623623623"
```

Lithnet Password Protection for Active Directory

P@ssw0rdis@s3cr3t!

Lithnet Password Protection for Active Directory provides flexible rules beyond that possible with group policies alone and prevents the use of previously compromised passwords. By Christian Knermann

Multifactor authentication (MFA) is the state of the art for securing user accounts and has long been recognized as such, ultimately even by users who are less IT savvy, now that numerous online services offer or even enforce MFA procedures. One service that many users encounter on a daily basis, however, usually only supports the traditional method of username and password: The security of an Active Directory infrastructure is defined by user account passwords. The free Lithnet Password Protection for Active Directory (LPP) provides more flexible rules than would be possible with group policies alone and prevents the use of previously compromised passwords. In this article, I look into how to commission and use LPP.

Length vs. Complexity

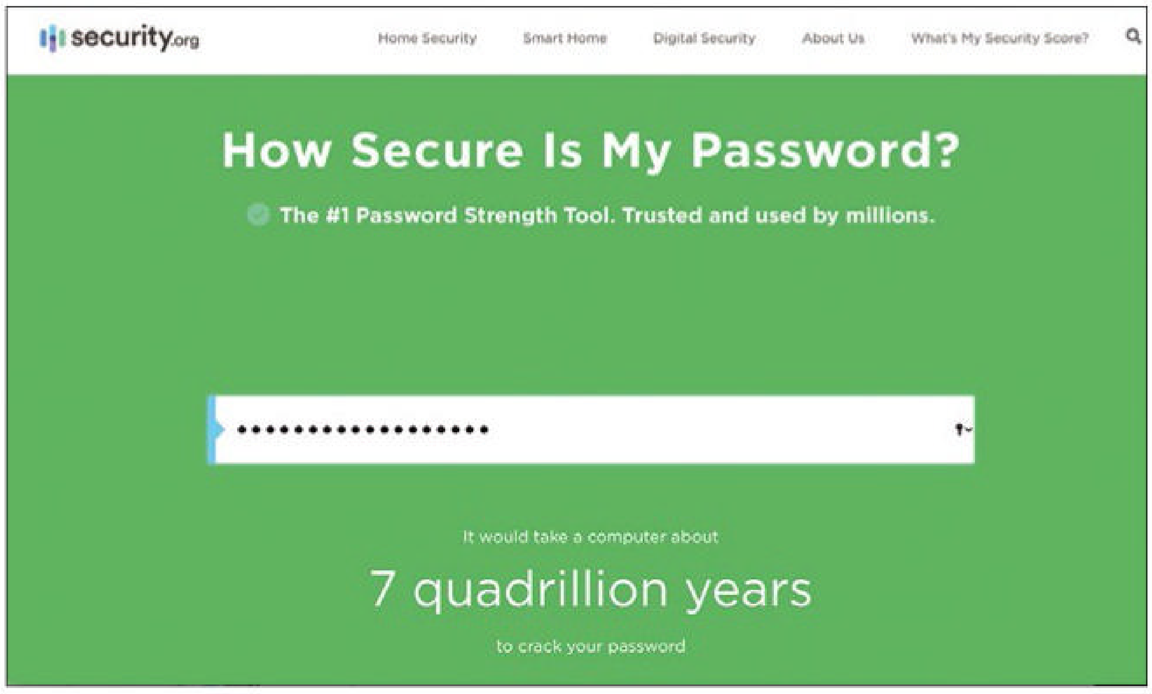
What constitutes a secure password and how often it should be changed is hotly debated among IT security experts worldwide. The consensus is that complexity and length are the decisive factors. The German Federal Office for Information Security (BSI) compares the two factors in its guidelines [1]. For example, the BSI recommends a high level of complexity for short passwords with a length of only eight to 12 characters. This typically means using four character types, of which many users will be familiar: a mix of upper- and lowercase letters, numbers, and special characters. The recommendations also lower the complexity requirements as the length increases. A significantly longer password with 20 to 25 characters may only have to meet two of the four complexity requirements. Indeed, the US National Institute of Standards and Technology (NIST), in the 2021 update to password guidance determined that length, "character for character," was more important than complexity [2].

The computational effort required to crack a password increases exponentially with password length. From 15 to 20 characters, cracking is no longer possible in a finite amount of time – or at least not with today's technology. However, length alone is not helpful if a password is susceptible to dictionary attacks or is found on a list of already compromised passwords. A certain degree of complexity is therefore advisable even for longer passphrases. The 18-character password from the title of this article, for example, would require a computer about 7 quadrillion years to crack, according to the *How Secure is My Password* site [3] (Figure 1).

Inflexible Group Policies

The onboard tools that Microsoft provides with Active Directory (AD) in group policies are not the best fit for implementing the previous considerations. The default settings for password security can be found in the Group Policy Object (GPO) of the *Computer Configuration | Policies | Windows Settings | Security Settings | Password Policies | Default Domain Policy*. A freshly installed domain controller on Windows Server 2022 sets the maximum password age to 42 days. In recent publications, the BSI and NIST, on the other hand, abandoned

Lead Image © Nah Ting Feng, 123RF.com



security.org

Home Security Smart Home Digital Security About Us What's My Security Score?

How Secure Is My Password?

The #1 Password Strength Tool. Trusted and used by millions.

••••••••••••••••••

It would take a computer about

7 quadrillion years

to crack your password

Figure 1: The Security.org website has a tool that tests the strength of your password.

recommending a regular password change. Therefore, you can decide for yourself whether to extend this period or just abolish it.

If you prefer a periodic change, the Default Domain Policy still defines a minimum password age of one day. This setting is intended to prevent resourceful users from changing their password several times in quick succession to restore the original value. The system actively prevents this by storing a history of the last 24 passwords used for each user.

The settings of the group policies in terms of complexity and length prove to be fairly inflexible. With Windows Server 2022, Microsoft sets the length to at least 7 characters out of the box and additionally enables the *Password must meet complexity requirements* setting, which ensures that passwords must contain three out of four possible character types – uppercase letters, lowercase letters, numbers, and symbols. A variable system that rewards people who use particularly long passwords with fewer complexity requirements is not something that Group Policy can inherently implement. However, the Microsoft password filters do offer third-party providers an interface for retrofitting functions [4].

Safer with Lithnet

Open source and free of charge, LPP docks onto the Group Policy interface when installed on domain controllers [5]. LPP comes with settings that relate complexity to password length, and it also compares passwords on demand against an admin-maintained list of blocked words, as well as the database of the *Have I been pwned?* (HIBP) service [6]. (The word “pwned” is a corruption of the term “owned.”) HIBP answers the question of whether a particular password is already on lists of stolen access data circulating on the Internet.

To configure LPP, you can use group policies. The password database is local, so the whole thing works without online access. Neither passwords nor hash values leave your internal network. The prerequisite for checking passwords against the HIBP list is that you first download and extract them in NTLM format (ordered by hash). This process takes a while because the list is 8.5GB packed and expands to 20GB unpacked. However, you only need this space temporarily. As soon as you import the list into LPP’s file-based database format, the space requirement drops again to 6GB.

Providing an HIBP Database

If you use multiple domain controllers, which is generally recommended in production operations, you have to decide where to store the database with the compromised passwords. Lithnet provides three strategies from which to choose: a shared repository in the form of a file share, local storage on each individual domain controller with manual replication by Robocopy or XCopy, and local storage with automatic replication by Microsoft’s Distributed File System (DFS).

Because HIBP’s list rarely changes, manual replication is a viable option for smaller environments. The manufacturer recommends DFS especially for geographically distributed infrastructures, explicitly pointing out that you should not misuse the existing SYSVOL share replication but create a dedicated DFS replication group.

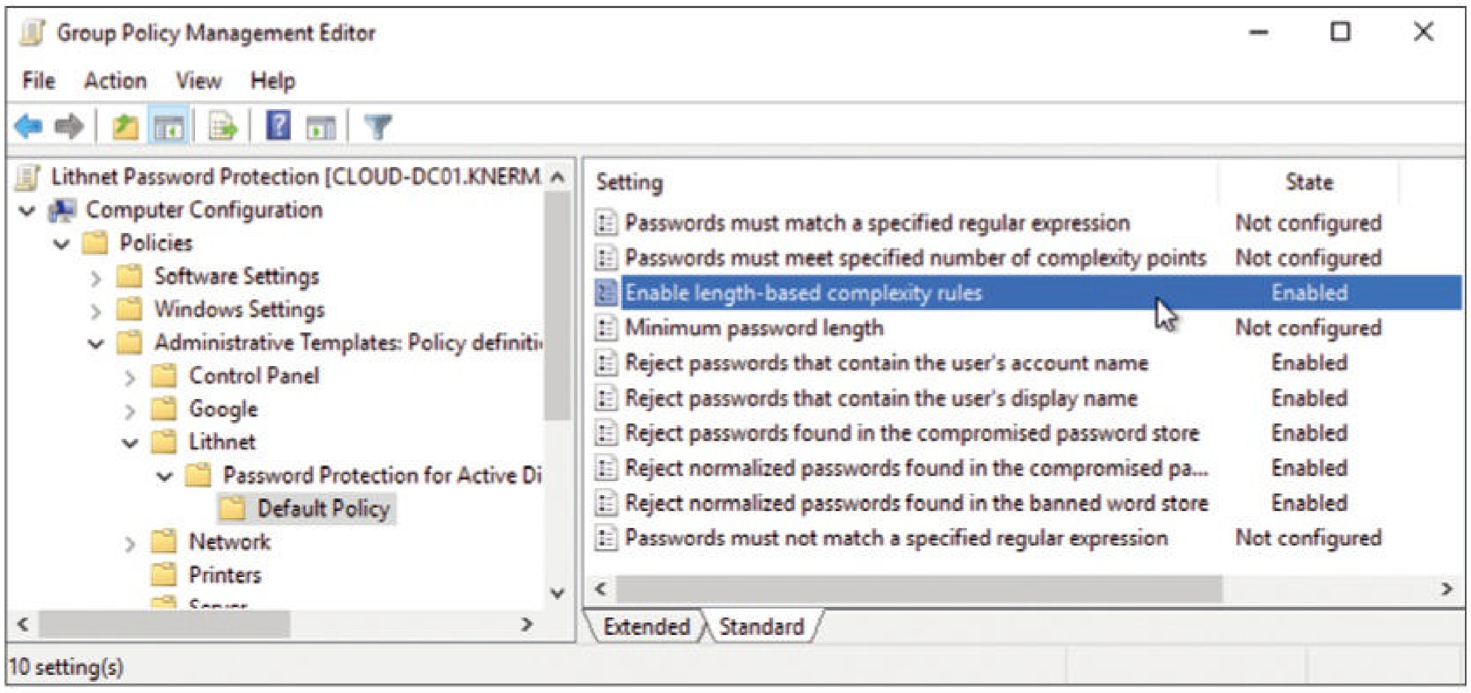
Once you have decided where the database will be located, download the LPP installation package and install the software in the context of a domain administrator, putting all components on each domain controller. In the second step of the dialog, proceed to configure the path to the database. The installation requires a reboot to enable the password filter. The following PowerShell commands,

```
Import modules LithnetPasswordProtection
Open-Store 'C:\Program Files\Lithnet\↲
  Active Directory Password Protection\↲
  Store'
Import-CompromisedPasswordHashes ↲
  -Filename C:\temp\pwned-passwords-ntlm-↲
            ordered-by-hash-v4.txt
```

use a PowerShell module provided by Lithnet to convert the HIBP list to an LPP database on a domain controller.

Preparing Group Policies

While this process is running, use the time to make further preparations. If you use a central repository for Group Policy templates in your domain, and this is the best choice for multiple domain controllers, then copy the two administrative templates (ADMX files) `lithnet.activedirectory.passwordfilter.admx` and `lithnet.admx` on a domain controller where LPP is installed from the local `C:\Windows\PolicyDefinitions` path to the central location: `\\<domain-name>\SYSVOL\<domain-name>\Policies\PolicyDefinitions`. Drop the associated



Setting	State
Passwords must match a specified regular expression	Not configured
Passwords must meet specified number of complexity points	Not configured
Enable length-based complexity rules	Enabled
Minimum password length	Not configured
Reject passwords that contain the user's account name	Enabled
Reject passwords that contain the user's display name	Enabled
Reject passwords found in the compromised password store	Enabled
Reject normalized passwords found in the compromised pa...	Enabled
Reject normalized passwords found in the banned word store	Enabled
Passwords must not match a specified regular expression	Not configured

Figure 2: LPP group policies allow far more granular settings than Microsoft's onboard tools.

ADML files into the appropriate subfolders, or at least populate the `en-US` folder.

Now you want to open Group Policy Management and, under *Default Domain Policy*, disable Microsoft's defaults for password length and complexity, because LPP will take care of that in a moment. Now create a new Group Policy Object (GPO), which you need to link at the top level of the domain or with the organizational unit (OU) of the domain controllers.

Next, open the new GPO in the Group Policy Management Editor and navigate to *Computer Configuration* | *Policies* | *Administrative Templates* | *Lithnet* | *Password Protection for Active Directory* | *Default Policy*. This folder contains the settings for configuring LPP (Figure 2). In the first step, enable the *Reject passwords found in the compromised password store* setting and check the options *Enable for password set operations* and *Enable for password change operations*.

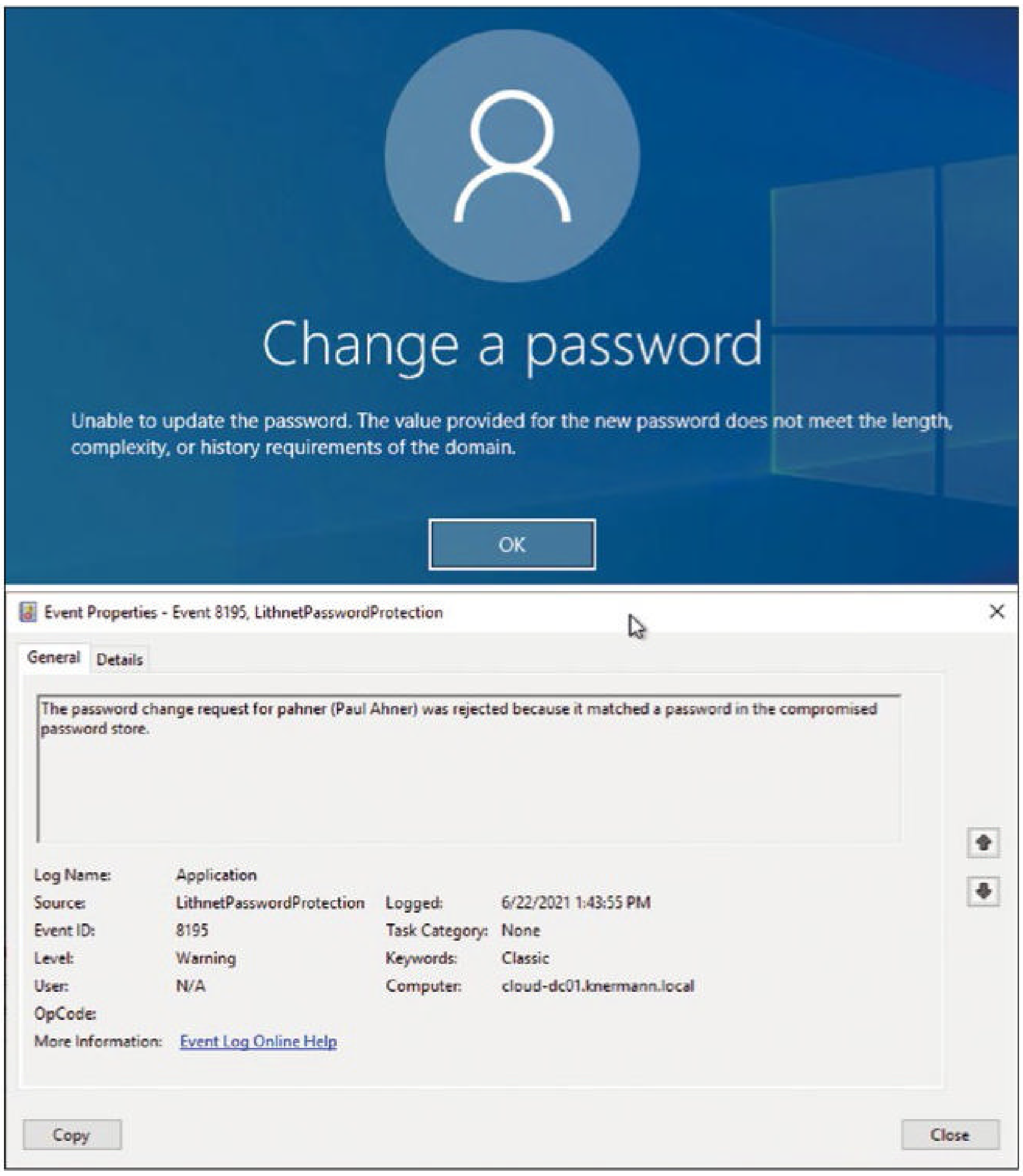
Few Meaningful Messages

You can wait for the Group Policy to update on all domain controllers, or you can enforce it with

```
gpupdate /force
```

In the next step, working in the context of any user on a client machine, try setting a password that is blacklisted by HIBP (e.g., *P4ssw0rd!*). You should get two results: LPP basically works and refuses to implement the change, and Windows acknowledges this action with the typical, uninformative default message.

Although LPP supports detailed password requirements under the hood, it unfortunately cannot influence client behavior. Therefore, end users don't find out what exactly caused their password change to fail. Administrators can discover the reason, though, in the *Application* event log of the domain controller that processed the password change attempt. LPP logs in detail which group policy setting intervened (Figure 3). All event IDs


Change a password

Unable to update the password. The value provided for the new password does not meet the length, complexity, or history requirements of the domain.

The password change request for pahner (Paul Ahner) was rejected because it matched a password in the compromised password store.

Log Name: Application
Source: LithnetPasswordProtection Logged: 6/22/2021 1:43:55 PM
Event ID: 8195 Task Category: None
Level: Warning Keywords: Classic
User: N/A Computer: cloud-dc01.knermann.local
OpCode:
More Information: Event Log Online Help

Figure 3: The local message on a Windows client is not very informative, but the event log on a domain controller reveals why a password change failed.

that you might encounter are documented [7].

Two settings reject passwords that contain the user's account or display name, thus preventing users from including their own names in the password, regardless of the password length and complexity. Regular expression fans can also look forward to two settings that allow or reject passwords on the basis of freely definable rules.

The *Passwords must meet specified number of complexity points* option provides you with an individually configurable system that allows you to specify a minimum number of points that a password must meet, as well as a number of points per uppercase letter, lowercase letter, number, and symbol. A user then accumulates approximately one point per uppercase letter, two for each number, and three for each special character. LPP only accepts the password if it reaches the minimum total number of points.

Blocking Individual Words

LPP's filter becomes even stricter if you enable the *Reject normalized passwords found in the compromised password store* and *Reject normalized passwords found in the banned word store* settings. For these settings to work, you need to feed the database of words to be blocked with some candidates. To do so, type e.g.:

```
Import modules LithnetPasswordProtection
Add-BannedWord Knermann
```

If you want to exclude a larger set of terms, you don't have to enter them manually. Instead, you can use the `Import-BannedWords` cmdlet to load entire dictionaries. The cmdlet expects a text file that contains one banned word per line.

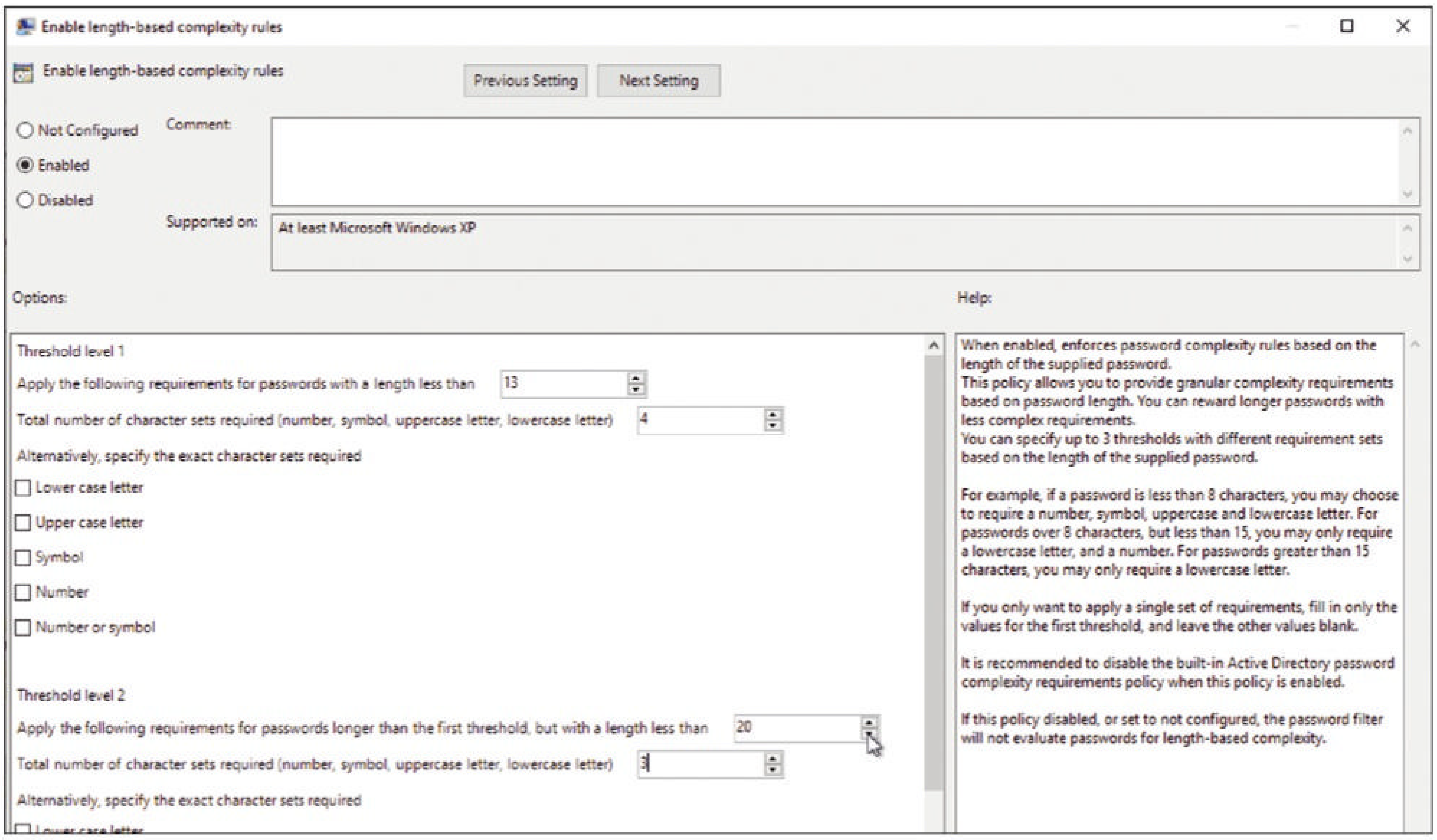
Normalization Increases Difficulty

When a user now enters a new password, LPP normalizes it before matching it against the blacklists. Normalization means that the filter first converts a password completely into lowercase letters and also removes spaces, numbers, and special characters at the beginning and end. Furthermore, LPP also understands "Leetspeak" (i.e., the creative replacement of letters with similar-looking numbers and symbols), which is popular on the Internet. For example, the filter normalizes the string "Kn3rm@nn!" to "knermann" and rejects the password change.

You can draw your own conclusions on the effectiveness of the filter rules by trial and error on a client computer, but you might want to use PowerShell instead. The `Get-PasswordFilterResult` cmdlet lets you test passwords against your set of rules and receive immediate, meaningful feedback on whether and why LPP rejects a particular password.

Implementing Recommendations

Enabling the *Minimum password length* setting enforces a minimum length of eight characters. You can now establish a direct link between length and complexity with the *Enable length-based complexity rules* option. You can define up to three threshold values for the length, each with different complexity requirements. In this way, you can implement the latest



Enable length-based complexity rules

Enable length-based complexity rules

○ Not Configured
● Enabled
○ Disabled

Supported on: At least Microsoft Windows XP

Threshold level 1

Apply the following requirements for passwords with a length less than 13

Total number of character sets required (number, symbol, uppercase letter, lowercase letter) 4

Alternatively, specify the exact character sets required

□ Lower case letter
□ Upper case letter
□ Symbol
□ Number
□ Number or symbol

Threshold level 2

Apply the following requirements for passwords longer than the first threshold, but with a length less than 20

Total number of character sets required (number, symbol, uppercase letter, lowercase letter) 3

Alternatively, specify the exact character sets required

□ Lower case letter

When enabled, enforces password complexity rules based on the length of the supplied password.
This policy allows you to provide granular complexity requirements based on password length. You can reward longer passwords with less complex requirements.
You can specify up to 3 thresholds with different requirement sets based on the length of the supplied password.

For example, if a password is less than 8 characters, you may choose to require a number, symbol, uppercase and lowercase letter. For passwords over 8 characters, but less than 15, you may only require a lowercase letter, and a number. For passwords greater than 15 characters, you may only require a lowercase letter.

If you only want to apply a single set of requirements, fill in only the values for the first threshold, and leave the other values blank.

It is recommended to disable the built-in Active Directory password complexity requirements policy when this policy is enabled.

If this policy disabled, or set to not configured, the password filter will not evaluate passwords for length-based complexity.

Figure 4: LPP rewards particularly long passwords with lower complexity requirements.

security recommendations and reward users of particularly long passwords with less stringent requirements with regard to password complexity.

For example, you might want to set *Threshold level 1* to 13 characters (Figure 4). In the next field, define how many of the four requirements – uppercase letters, lowercase letters, numbers, symbols – a password with fewer characters needs to meet. By way of an example, I selected the highest value: four out of four. Alternatively, you can use checkboxes to specify exactly which of the requirements the password must meet.

You can set *Threshold level 2* to 20 characters and specify that passwords with fewer characters must still meet three of the four requirements. The lowest range of the configuration applies to all passwords longer than the second threshold. In this case, if you still require two of the four requirements and apply the policy, LPP enforces the password filter to match the BSI recommendations.

Testing Existing Passwords

The rules of the game defined by GPO immediately affect any attempts by users to change their password. Even an admin setting a new password for a user account in the Active Directory Users and Computers console cannot override the rules. What about passwords that existed before you established the set of rules for LPP, though? This is where the `Test-IsADUserPasswordCompromised` cmdlet can help. For a single user account, you can run it to check whether it uses a password known to be compromised:

```
Test-IsADUserPasswordCompromised ⮐
  -UPN <paul.ahner@knermann.local>
```

The cmdlet returns True or False. You can test all the user accounts in your environment in one fell swoop with a PowerShell script provided by Lithnet [8]. It writes all user accounts with blank passwords or passwords on the HIBP list to a CSV file so you can inform the affected users and prompt them to change their passwords.

Note that the cmdlet reads the hash value of the password from the AD database, not the password itself. Consequently, the cmdlet can only check whether the password is known in HIBP. The tool does not determine whether the password meets your other complexity requirements. Furthermore, reading a hash value also requires very far-reaching permissions. The account in whose context you use the cmdlet and the script must be a member of the Domain Admins group or have at least the *Replicating Directory Changes All (DS-Replication-Get-Changes-All)* permission, which is popular among users of the Mimikatz tool, which can be abused in a DCSync attack [9]. Accordingly, you should only use the tool and script on trusted devices, preferably directly on a domain controller.

Conclusions

Lithnet Password Protection for Active Directory proves to be far more flexible than Microsoft's onboard tools. Unlike the onboard tools, LPP supports variable rules that check the length of a password as a function of its complexity. The only disadvantage is that LPP does not improve the Windows client's ability to communicate problems. If a password does not pass your set of rules, Windows will only return an uninformative default message to the end user with no indication of exactly which condition prevented the password from being changed. The introduction of LPP needs to be accompanied by an organizational campaign to make your users aware of the need for secure passwords and an explanation of the new rules. ■

Info

[1] BSI: Creating secure passwords: [https://www.bsi.bund.de/DE/Themen/Verbraucherinnen-und-Verbraucher/Informationen-und-Empfehlungen/Cyber-Sicherheitsempfehlungen/Accountschutz/Sichere-Passwoerter-erstellen/sichere-passwoerter-erstellen_node.html] (in German)

[2] NIST Password Recommendations: [https://www.netsec.news/summary-of-the-nist-password-recommendations-for-2021/]

[3] How secure is my password?: [https://www.security.org/how-secure-is-my-password/]

[4] Password filter: [https://docs.microsoft.com/en-us/windows/win32/secmgmt/password-filters]

[5] Lithnet Password Protection: [https://github.com/lithnet/ad-password-protection]

[6] Published passwords: [https://haveibeenpwned.com/Passwords]

[7] Lithnet event logging and reporting: [https://github.com/lithnet/ad-password-protection/wiki/Event-logging-and-reporting]

[8] Audit of existing passwords: [https://github.com/lithnet/ad-password-protection/wiki/Audit-existing-passwords]

[9] DCSync attacks against the Active Directory: [https://hkeylocalmachine.com/?p=928]

Author

Christian Knermann is Head of IT-Management at Fraunhofer UMSICHT, a German research institute. He's written freelance about computing technology since 2006.

Public Money Public Code

Modernising Public Infrastructure with Free Software

fsfe Free Software Foundation Europe

Learn More: https://publiccode.eu/

Security features in Windows Server 2022

Fine Tuning

The release of Windows Server 2022 adds some new security features to its server operating system that might not be earth-shattering; however, you will find Secured-core, DNS over HTTPs, TLS 1.3, and Azure Stack HCI genuinely useful in your constant fight to harden server operations. By Thomas Joos

Compared with Windows Server 2019, spectacular changes to Windows Server 2022 are few and far between. However, the sum total of improvements make the latest version a more secure option than its predecessors. For example, the 2022 release comes with new Group Managed Service Accounts (gMSAs) for Windows containers without having to add the host to the domain. This change enhances security for container hosts that you do not want to be part of your Active Directory.

If you deploy Windows Server 2022 on Microsoft Azure, you can select the images on which Azure security policies are automatically enabled, which is clear evidence of Microsoft's focus on security with its new operating system. Another factor is replacing the aging Internet Explorer on servers. Windows Server 2022 comes with the state-of-the-art Edge browser preinstalled by default in the core.

Beyond these changes, Microsoft's new Security Baseline for Windows Server 2022 ensures superior protection with additional security settings and recommendations delivered by group policies. In combination with Windows 11, Windows Server 2022 offers greater security than the combination of Windows 10 and Windows Server 2019, which makes it worth your while to take a closer look at the opportunities these innovations present.

Secured Core

Windows Server 2022 also sees Microsoft introduce the Secured-core server, which, in simple terms, is the security standard for a Windows Server, wherein the operating system optionally uses the hardware functions for greater security, and conversely, the server hardware is precisely designed for Windows Server 2022. Secured-core server gives the enterprise a coherent combination of hardware, drivers, software, and Windows Server 2022.

Secured-core uses hypervisor-protected code integrity (HVCI), kernel direct memory access (DMA) protection, SystemGuard, Secure Boot, virtualization-based security (VBS), and Trusted Platform Module (TPM) 2.0 security features. These technologies must be present and enabled on the server. Customers who purchase a Secured-core server in cooperation with a hardware manufacturer are sure to receive server hardware capable of handling the required feature set.

If Windows Server 2022 is not yet configured to support the Secured-core features, the features can be set up centrally in the Windows Admin Center (WAC). You can see the individual mandatory items for Windows Server 2022 as a Secured-core server and enable the security features there. The new version of the WAC *Security* extension is essential for admins wanting to manage the hardware aspects of the Secured-core. It comes with a new Secured-core menu item (Figure 1) that you use to determine whether your hardware supports the individual features and whether they are enabled on your Windows Server 2022. To unhide the extension, type the feed address *https://aka.ms/wac-insiders-feed* under *Extensions | Feeds* in the Admin Center settings. After doing so, the extensions will be updated to give you the new version of *Security*.

If your server hardware does not support individual parts of Secured-core

Photo by Matthew Henry on Unsplash

or if they are not enabled in the Unified Extensible Firmware Interface (UEFI)/BIOS, the Windows Admin Center displays a *Not supported* message. If the hardware supports the Secured-core feature in question, *Not configured* is displayed for features that are installed on the server but not enabled. You can then use Windows Admin Center to enable these functions and proceed to configure them as needed. Conveniently, tool tips display when you mouse over the various security features.

DNS over HTTPS

Client-side support for secure name resolution – DNS over HTTPS (DoH) – is another innovation in Windows Server 2022, and you will find it in Windows 11, too. You can configure this in Windows Server 2022 and Windows 11 under *Settings* | *Network & Internet* | *Ethernet*. The options *DNS settings* and *Edit* are found in the network adapter settings.

You can also use group policies to define the settings, configured under *Computer configuration* | *Administrative templates* | *Network* | *DNS Client* | *Configure DNS over HTTPS (DoH) name resolution*. A word of warning: If you use group policies to enable DoH in Windows Server 2022, but the DNS servers you use do not support DoH, name resolution will stop working.

The encrypted connection between the DNS client and DNS server protects queries against attacks. That said, both Windows Server 2022 and Windows 11 only support DoH as clients. In other words, you cannot use Windows Server 2022 as a secure DoH server. You have three options in the settings for connecting clients to DNS servers (Figure 2):

- *Unencrypted only*: The client with Windows 11 and Windows Server 2022 does not use DoH encryption for DNS queries.
- *Encrypted only (DNS over HTTPS)*: The client exclusively uses encrypted connections for resolution with DNS. If no secure connection to the DNS server is available, no name resolution takes place.
- *Encrypted preferred, unencrypted allowed*: The client allows the use of encrypted connections but also uses unencrypted connections, if required.

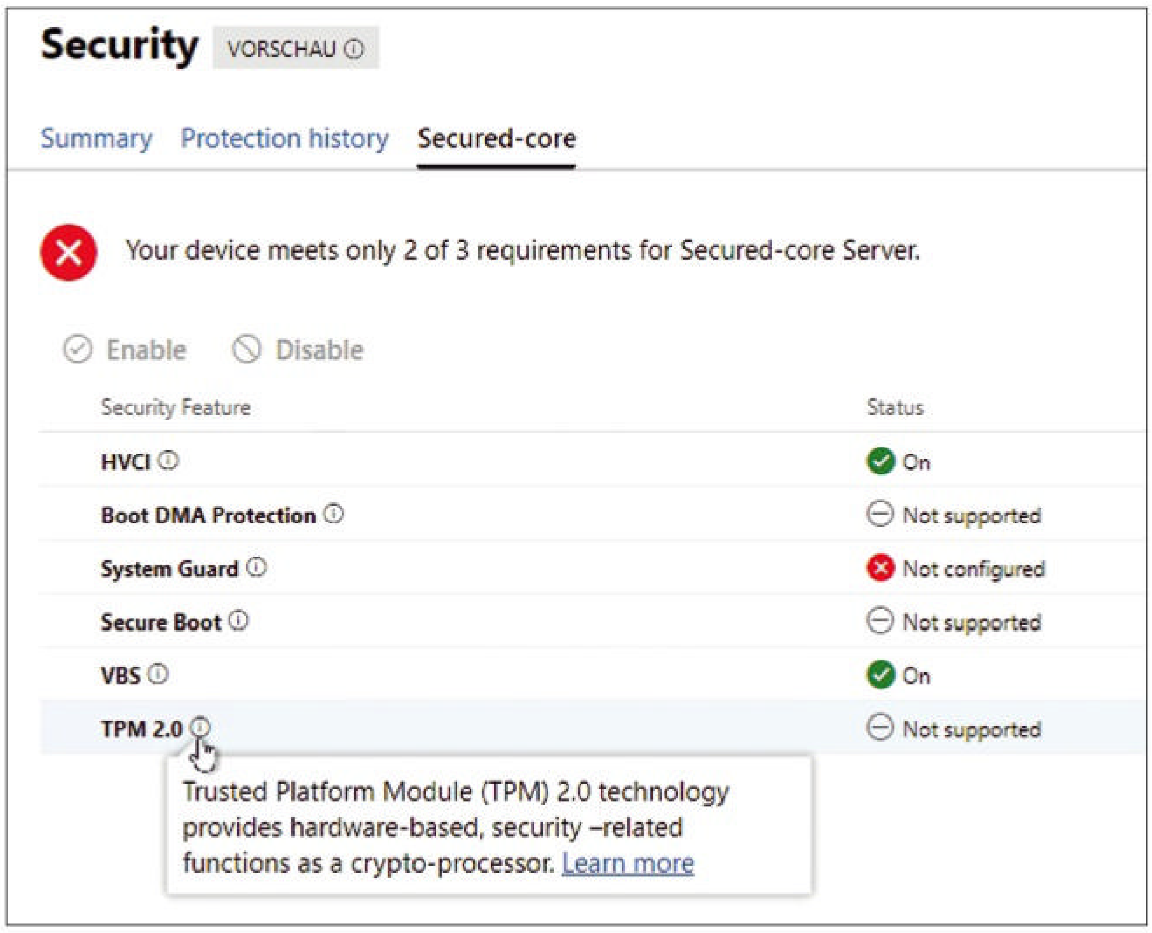
The options are only available if the specified DNS server supports the functions and is stored as such on Windows Server (e.g., servers with IP addresses 1.1.1.1 (Cloudflare) and 8.8.8.8 (Google)). In Windows Server 2022 and Windows 11 PowerShell, you can use the `Get-DNSClient-DohServerAddress` cmdlet to view the currently supported DNS servers. To add new servers, use:

```
Add-DnsClientDohServerAddress ↵
  -ServerAddress <IP address> ↵
  -DohTemplate <template> ↵
  -AllowFallbackToUdp $False ↵
  -AutoUpgrade $True
```

You can also use the name resolution policy table (NRPT) at this point to configure queries to a DNS namespace for use of a static DNS server. In this case, encrypted DNS connections can be used for certain queries.

Protected Data Transfer

Windows Server 2022 supports Server Message Block (SMB) protocol encryption with AES 256 GCM (Galois/counter mode) and CCM (AES 256 counter mode and cipher block chain MAC (message authentication code)). HTTPS and TLS 1.3 are the defaults


Security VORSCHAU ⓘ

Summary | Protection history | Secured-core

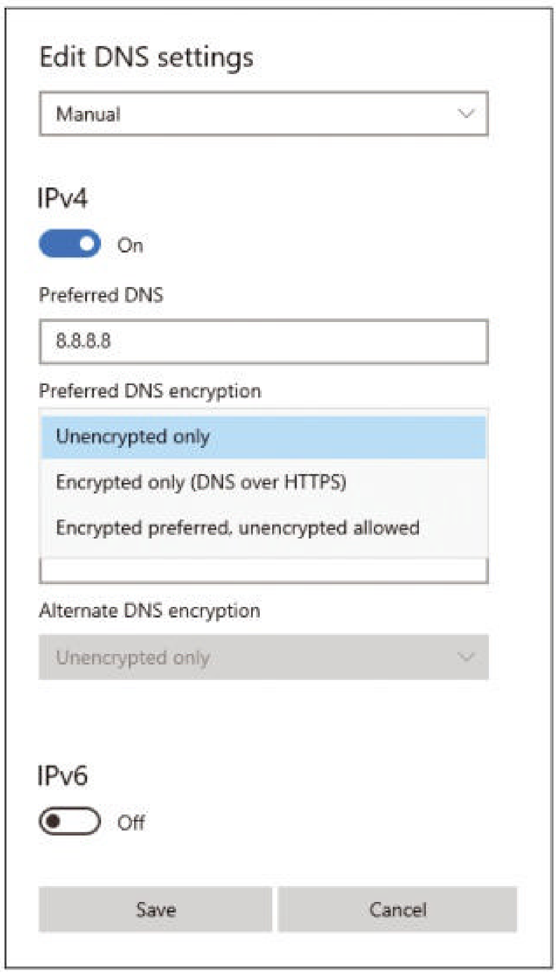
Your device meets only 2 of 3 requirements for Secured-core Server.

Enable | Disable

Security Feature	Status
HVCI ⓘ	✓ On
Boot DMA Protection ⓘ	⊖ Not supported
System Guard ⓘ	✗ Not configured
Secure Boot ⓘ	⊖ Not supported
VBS ⓘ	✓ On
TPM 2.0 ⓘ	⊖ Not supported

Trusted Platform Module (TPM) 2.0 technology provides hardware-based, security –related functions as a crypto-processor. Learn more

Figure 1: Secured-core functions can be managed in the Windows Admin Center.


Edit DNS settings

Manual

IPv4

On

Preferred DNS: 8.8.8.8

Preferred DNS encryption:
- Unencrypted only
- Encrypted only (DNS over HTTPS)
- Encrypted preferred, unencrypted allowed

Alternate DNS encryption: Unencrypted only

IPv6

Off

Figure 2: DNS over HTTPS in Windows Server 2022 secures name resolution, although it must be supported by the DNS server.

in Windows Server 2022. When clients connect to the server, the server tries to use HTTPS and TLS 1.3, if possible.

Cluster nodes in Windows Server 2022 also support encryption for internal communication, as is the case for access to Cluster Shared Volumes (CSVs) and for communication in Storage Spaces Direct. Encryption technologies also take effect in Remote Direct Memory Access (RDMA) and SMB Direct. Windows Server 2022 uses AES 128 and AES 256, and the connection variants and encryption technologies that offer maximum security are always enabled by default. This setup is intended to improve security for network traffic. The use of Windows 11 or at least Windows 10 21H2 is ideal in this scenario.

Hotpatching in the Cloud

Windows Server 2022 Datacenter Azure Edition is a new variant that can only be used on Azure and Azure Stack hyperconverged infrastructure (HCI). In another part of this article, I highlight the advantages of Azure Stack HCI on the network. In version 21H2, the on-premises installation of Azure also uses features from Windows Server 2022. The current Azure Stack HCI 20H2 version still relies on Windows Server 2019 as the base operating system.

However, Microsoft has also clarified that Azure Stack HCI does not simply rely on Windows Server as the base operating system but actually extends its feature set. Azure Stack HCI 21H2 integrates features that did not make the cut for Windows Server 2022, including various optimizations for Storage Spaces Direct. Azure Stack HCI for HCI volumes includes thin provisioning; this technology is not on board in Windows Server 2022 Datacenter. Support for physical graphics cards for virtual machines (VMs) is also integrated in Azure Stack HCI but is not available in Windows Server 2022. If you rely on clusters on the network, you might want to at least consider using Azure Stack HCI if you migrate to Windows Server 2022.

Stretched clustering (i.e., clusters spanning multiple data centers) is now only possible with Azure Stack HCI. Extended Security Updates, meanwhile, extend support for Microsoft operating systems. These capabilities are only available for VMs on Azure Stack HCI and in Microsoft Azure. These examples are just a few of the feature differences and are interesting for corporations that still rely on Windows Server 2012/2012 R2 or SQL Server 2012. Support for these systems will expire in 2022 or 2023. By the way, Azure Stack HCI billing uses a pay-per-use model.

Another interesting thing from a security perspective is that the Azure Edition supports hotpatching. This technology removes the need to reboot the entire server after an update and, instead, only restarts individual areas of the kernel and the operating system. This simplifies the installation of updates and keeps the server available to users without downtime. Restarting individual elements of the operating system, for example, does not cause workload failures and users will hardly notice a thing. If you want to use the server locally on the network, you can also deploy this edition on Azure Stack HCI 21H2.

SMB over Quick UDP Internet Connections (QUIC) protocol is also a feature only available in Windows Server 2022 Datacenter Azure Edition. Clients use the QUIC protocol for communication rather than TCP. In combination with TLS 1.3, applications can access data in a far more secure way, especially where servers are located on edge networks.

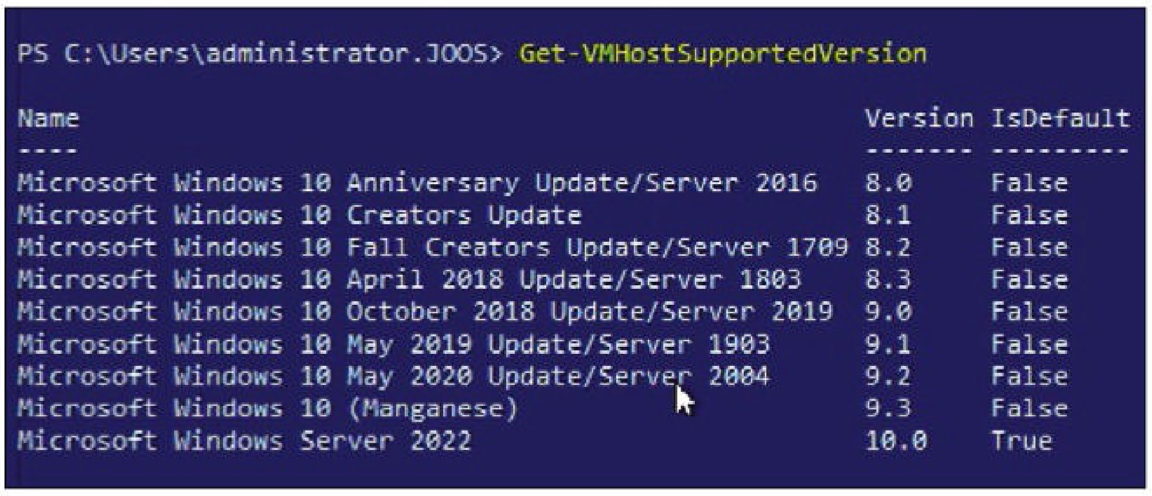
HCI Instead of Shielded VMs

Hyper-V does not have as many innovations as Windows Server 2019, but the changes are still quite significant in terms of security. Again, Windows 11 and Windows Server 2022 are quite similar in terms of innovations. The new features in Hyper-V are also available in Azure Stack HCI 21H2 but are not included in Azure Stack HCI 20H2.

A free Hyper-V server for Windows Server 2022 is no longer available. The last free version is Hyper-V Server 2019. According to the official recommendation, companies looking for a standalone server for virtualization should go for Azure Stack HCI. The downside is that this edition is not available for free. Azure Stack HCI sees Microsoft offer its own operating system, which takes Azure functions and an HCI infrastructure to the data center. Just like Windows Server 2022, Azure Stack HCI can be managed from the Windows Admin Center.

Microsoft has also removed shielded VMs from Windows Server 2022. Although this technology is still integrated in Windows Server 2022, it is no longer under active development and no longer recommended for operation. If you want a secured fabric, you can turn to Azure Stack HCI, according to Microsoft. Microsoft's HCI environment is optimized for the secure operation of VMs.

Windows Server 2022 and Windows 11 see the introduction of a new version 10 for VMs. Windows 10 and Windows Server 2019 still use version 9. If you update a Hyper-V host



```
PS C:\Users\administrator.JOOS> Get-VMHostSupportedVersion

Name                                                  Version IsDefault
----                                                  ------- ---------
Microsoft Windows 10 Anniversary Update/Server 2016   8.0     False
Microsoft Windows 10 Creators Update                  8.1     False
Microsoft Windows 10 Fall Creators Update/Server 1709 8.2     False
Microsoft Windows 10 April 2018 Update/Server 1803    8.3     False
Microsoft Windows 10 October 2018 Update/Server 2019  9.0     False
Microsoft Windows 10 May 2019 Update/Server 1903      9.1     False
Microsoft Windows 10 May 2020 Update/Server 2004      9.2     False
Microsoft Windows 10 (Manganese)                      9.3     False
Microsoft Windows Server 2022                         10.0    True
```

Figure 3: PowerShell lets you view supported VM versions on Hyper-V hosts.

directly, the VMs will keep the previous version. On the host, you can use the Get-VMHostSupportedVersion cmdlet to check which versions a host supports (Figure 3). The versions of each VM can be seen in Hyper-V Manager and also in the Windows Admin Center. PowerShell lets you view the version of each VM with the cmdlet:

```
Get-VM * | format-table name, version
```

To switch to the new version, use the command:

```
Update VMVersion <name of VM>
```

For outdated versions, the *Upgrade Configuration Version* option is available in the contextual menu of the VM in Hyper-V Manager. When creating VMs in PowerShell, you can also control the edition by typing, for example:

```
New-VM -Name "WindowsCV9" -Version 9.0
```

Now comes nested virtualization: Windows Server 2022 and Windows 11 support computers with AMD processors. Windows 10 and Server 2019 can only use computers with Intel processors for this technology. Embedded virtualization plays an important role for test and development environments, container hosts, and virtual clusters.

Virtual switches now support Receive Segment Coalescing (RSC) in Windows Server 2022. The switches can combine network packets and send them together. The data is unpacked again on the host or VM for which the segment is intended. Data traffic between virtual network adapters on the same host can also be controlled and optimized in this way, which speeds up network traffic and improves security while reducing the load on the network adapters and hardware of the computers involved. It also significantly reduces the load on the CPUs. Although Windows Server 2019 already supports RSC, Microsoft has significantly upgraded the technology in Windows Server 2022. The settings can be defined on each Hyper-V host. The commands for disabling and enabling are:

```
Set-VMSwitch -Name vSwitchName ⤶
             -EnableSoftwareRsc $false
Set-VMSwitch -Name vSwitchName ⤶
             -EnableSoftwareRsc $True
```

You can retrieve the status of the settings with the command:

```
Get-VMSwitch -Name vSwitchName | ⤶
  Select-Object *RSC*
```

If several network adapters are available on a Hyper-V host, Windows Admin Center automatically proposes that you create a switch-embedded teaming (SET) switch when you set up new virtual switches. Avoid working with the teaming functions in Windows Server 2022 up front and go for the SET options in the Admin Center instead.

Management with WAC

Microsoft has increasingly been moving features for managing clusters and Hyper-V to the Windows Admin Center, as is also the case for Windows Server 2022, so it makes sense to take a closer look at the WAC options when using Windows Server 2022. Of course, the setting options are still part of the Failover Cluster Manager, but using Windows Admin Center for management tasks typically makes more sense.

Under *Virtual machines*, you will find the *Affinity rules* settings, which is where you define how the cluster will position VMs. Windows Server 2022 and the Windows Admin Center offer very extensive options. Windows then stores the rules as a cluster object to keep them available at all times. For example, you can define which VMs will be positioned together on a host and which VMs the cluster will keep apart. Of course, these rules are also available in PowerShell, as are all functions available in the Windows Admin Center or the Failover Cluster Manager as a general rule.

Cluster validation tests are still used in Windows Server 2022 as part of the process of creating and operating a cluster. Microsoft has extended these in Windows Server 2022 to cover more Hyper-V features in the cluster in the tests, optimizing cluster operations and security with Windows Server 2022 and Hyper-V after installation. During the tests, the wizard also checks, for example, the teaming functions of the switches and the RDMA functions.

Conclusions

The latest addition to the Microsoft server family, Windows Server 2022, might not come with truly earth-shattering innovations. But a closer look reveals attention to detail that will offer organizations many benefits in day-to-day operations – naturally with a view to security, as well. ■

Chef users faced with a license change might find solace in a new open source distribution, Cinc.

Trap and Release

For the past two years, Chef software used commercially has required a separate license. Cinc enters the scene as a free and completely compatible Chef distribution that makes it a genuine alternative. By Martin Loschwitz

Administrators who primarily move around in the open source scene are not exactly euphoric when they are forced to deal with the details of software licenses. The Apache license has asserted itself in recent years, and there is still the good old GNU Public License (GPL), preferably version 2.

A common practice in the free/libre open source software (FLOSS) world is to distribute compiled programs under the same license as the original source code – or at least the overwhelming majority of open source projects does things this way.

One notable exception to this rule is Chef, which made a name for itself years ago as one of the first automators. Since the spring of 2019, however, the relationship between users and provider has been troubled. Anyone wanting to use prebuilt packages for Chef now needs a commercial license, which was not necessary before.

As you are probably aware, it doesn't take long in the open source world for a free alternative to take root. Cinc enters the fray with the promise of being fully compatible with Chef while remaining available under a free license. This article begins by explaining what the implications of the Chef licensing model change were and how Cinc is positioning itself as an alternative.

How It (Often) Works

The day that extensive changes to Chef licenses took effect was April 2, 2019 (Figure 1), directly affecting those Chef users who had been using Chef normally like any other FLOSS software.

Although it is not legally required and in fact not necessary to give away the results of compiling an open source software product for free, a certain standard approach for distributing software to the general public has established itself in the FLOSS world.

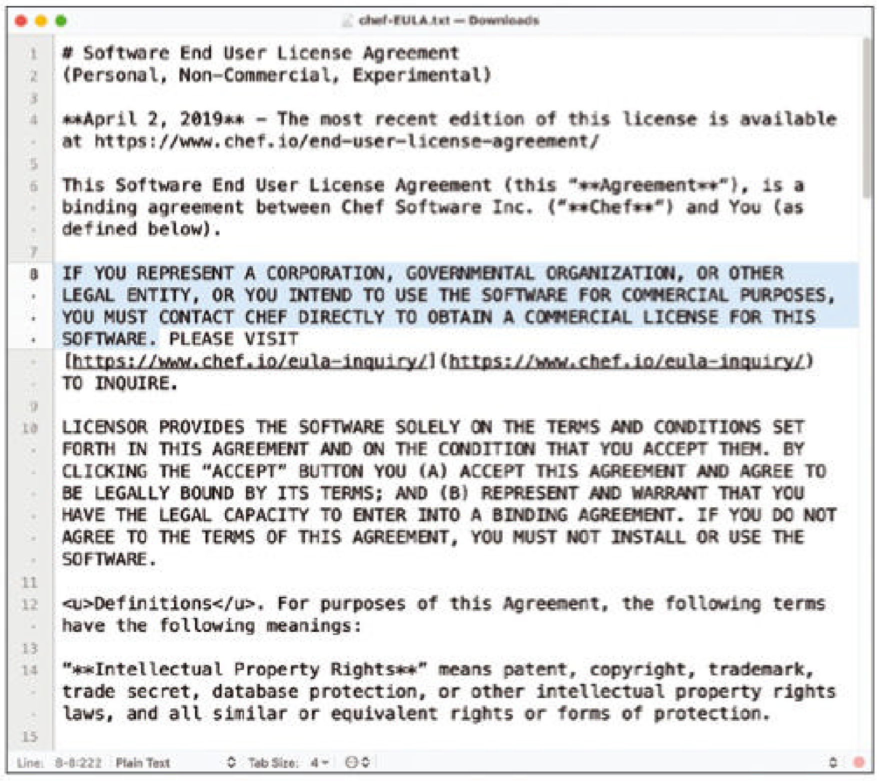
It works like this: Vendors such as Docker or projects such as PostgreSQL have a vested interest in providing their own user base with working versions of their software. In the past, these versions mostly took the form of packages: Docker, for example, still offers the Community Edition of its container environment, which runs on all common Linux distributions. The system administrator can easily use the packages offered by the vendor. To do so, they add the corresponding vendor repositories to the respective package managers on their systems and download the packages available there. In today's hip world of Docker and the like, containers that manufacturers offer instead of packages are often used for the same task. But the principle remains the same at its core.

This approach does have advantages for the providers. They regularly use the freely available packages of their own software as the basis for commercial extensions. Above all, however, they offer the manufacturer a certain degree of control over which version of their own product people use. It makes life more difficult to provide support and track bugs if every system administrator is working with self-compiled binaries. On top of that, software vendors can ensure that the packages used by users meet certain quality standards and are not deprecated. Chef and the company behind it, Progress Software, adhered to this very principle for quite a while – including proprietary tools whose source code was not publicly available.

Well Meant, Badly Done

Chef's licensing changes, which went into effect in April 2019, had two main goals. You can't find fault with the first: The vendor wanted to free itself from its closed-source components and put all of its proprietary tools under a free license – or at least

Photo by Pascal Bernardon on Unsplash



```
# Software End User License Agreement
(Personal, Non-Commercial, Experimental)

**April 2, 2019** - The most recent edition of this license is available
at https://www.chef.io/end-user-license-agreement/

This Software End User License Agreement (this “**Agreement**”), is a
binding agreement between Chef Software Inc. (“**Chef**”) and You (as
defined below).

IF YOU REPRESENT A CORPORATION, GOVERNMENTAL ORGANIZATION, OR OTHER
LEGAL ENTITY, OR YOU INTEND TO USE THE SOFTWARE FOR COMMERCIAL PURPOSES,
YOU MUST CONTACT CHEF DIRECTLY TO OBTAIN A COMMERCIAL LICENSE FOR THIS
SOFTWARE. PLEASE VISIT
[https://www.chef.io/eula-inquiry/](https://www.chef.io/eula-inquiry/)
TO INQUIRE.

LICENSOR PROVIDES THE SOFTWARE SOLELY ON THE TERMS AND CONDITIONS SET
FORTH IN THIS AGREEMENT AND ON THE CONDITION THAT YOU ACCEPT THEM. BY
CLICKING THE “ACCEPT” BUTTON YOU (A) ACCEPT THIS AGREEMENT AND AGREE TO
BE LEGALLY BOUND BY ITS TERMS; AND (B) REPRESENT AND WARRANT THAT YOU
HAVE THE LEGAL CAPACITY TO ENTER INTO A BINDING AGREEMENT. IF YOU DO NOT
AGREE TO THE TERMS OF THIS AGREEMENT, YOU MUST NOT INSTALL OR USE THE
SOFTWARE.

<u>Definitions</u>. For purposes of this Agreement, the following terms
have the following meanings:

“**Intellectual Property Rights**” means patent, copyright, trademark,
trade secret, database protection, or other intellectual property rights
laws, and all similar or equivalent rights or forms of protection.
```

Figure 1: Not good news for Chef users: After a license change, you are no longer allowed to use the compiled Chef packages for free in commercial operations.

the source code for them. Here’s the rub: At the same time as changing the license of the source code of various tools, Progress Software introduced an end-user license agreement (EULA). The manufacturer clarified that the EULA does not explicitly apply to the source code of the software but to the binary packages provided by the vendor. Anyone who had already compiled Chef themselves by then was practically unaffected by the changes.

The number of admins to whom this applies is probably very small. The number of system administrators that Progress Software tripped up with the license change is undoubtedly far larger. The manufacturer has tried to appease its clientele. The new EULA does not apply to all Chef components. The central tools of a Chef installation are still available as binaries and are under a free license.

If you have used Chef commercially so far – and commercially in the eyes of the vendor means in a scenario designed to make a profit – you are likely to have used the vendor’s binary packages to do so.

License Trickle-Down

From today’s perspective, Chef users are justified in seeing this announcement as doing lip service to the FLOSS ideals. While central tools such as the Chef Workstation, Infra Client, or Infra Server initially remained available as binaries under the usual licenses, the manufacturer has tightened the reins on later versions and made the new EULA mandatory there, too. Today, not a single one of the central Chef components is available in binary form under a free license.

Admins whose entire automation is based on Chef therefore face a dilemma. The Chef packages they deployed will gradually became unusable without being guilty of breaking the license agreement if they refuse to purchase a Chef license, because the new EULA only allows the use of the packages without a commercial agreement with Progress Software for experimental or personal purposes. It’s also conceivable that some admins haven’t even noticed the license change and are using Chef illegally as they read this article.

Difficult to Justify

By the way, the purported reason for the license changes, according to the manufacturer at that time, was to unify the license model of the software and offer the best possible support for customers who use Chef commercially. Progress Software also offered a questionable comparison: They claimed that, basically, the new business model is no different from the one that Red Hat and SUSE use for their distributions in the enterprise segment. Most of the software there is open source, too, but users have to pay for commercial use.

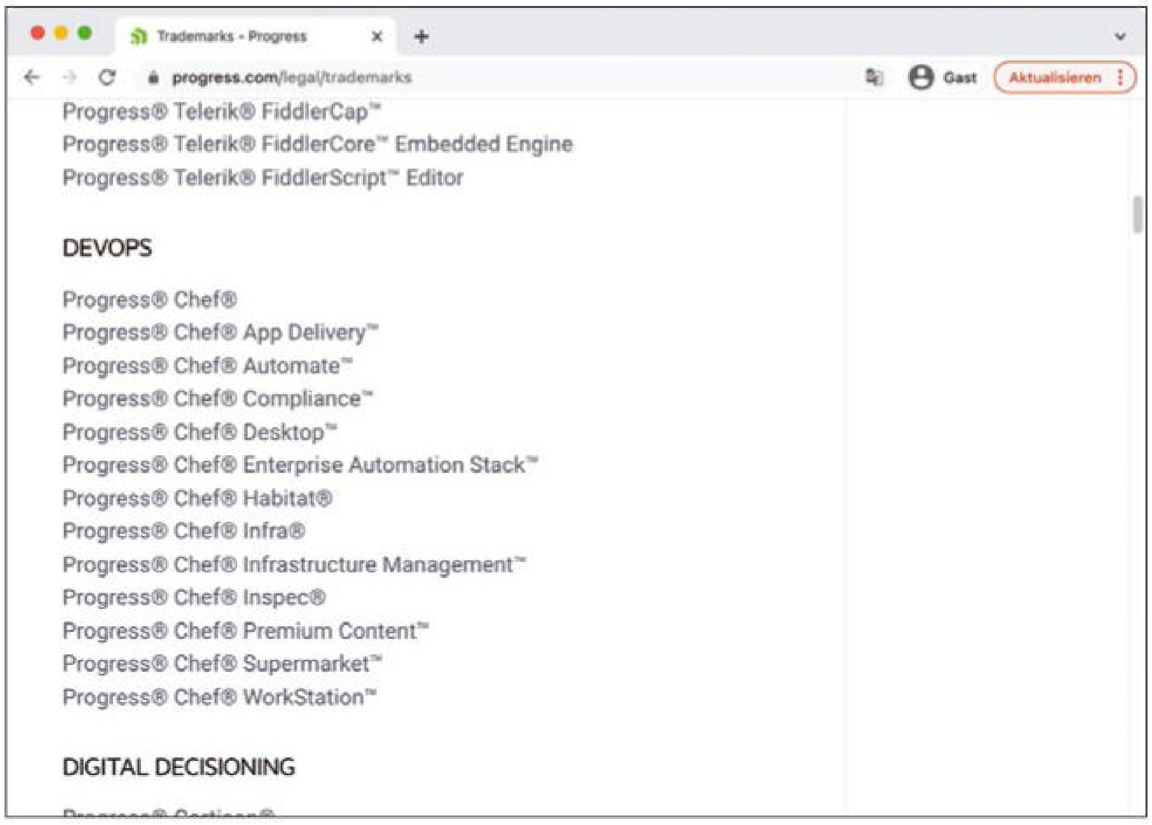
This comparison admittedly ignores the fact that users have had a binary-compatible distribution in the form of CentOS as an alternative to Red Hat since 2004 at the latest; this distribution was even maintained directly by Red Hat for many years. If you owned a Red Hat setup you could therefore switch to CentOS with relatively little overhead, and many admins have made extensive use of this option. How important CentOS is on the market was revealed when Red Hat decided to kill the distribution. Within a few hours, two projects were founded in the form of Rocky Linux and AlmaLinux to continue offering a fully RHEL-compatible distribution.

The situation is different for Chef administrators. They made a conscious decision to use the tool for automation a few years back and have since developed a complex tool set with this solution. The license change forcd them to make another decision: Either pay or use a different automation solution. Introducing a compliant solution would cost a great deal of money, as anyone with any expertise knows, but above all it would mean a massive workload – at a time when companies are already desperately looking for skilled personnel.

Trademarks and Other Blunt Weapons

To ensure that this pretty new licensing system is not endangered by interference from the sidelines, Chef also introduced a brand-new trademark hammer at the same time as the new licensing setup (Figure 2). Since then, only the Chef version that comes directly from Progress Software in the binary form can be referred to as Chef. The reasons given for this are as predictable as they are silly: Progress Software does not want to “dilute the brand” and it wants to “ensure that admins get the highest quality software when they install binaries that have Chef in their name.”

What this means in concrete terms is that distributors who compile Chef from the source code themselves (i.e., who build their own Chef distribution)



Progress® Telerik® FiddlerCap™
Progress® Telerik® FiddlerCore™ Embedded Engine
Progress® Telerik® FiddlerScript™ Editor

DEVOPS

Progress® Chef®
Progress® Chef® App Delivery™
Progress® Chef® Automate™
Progress® Chef® Compliance™
Progress® Chef® Desktop™
Progress® Chef® Enterprise Automation Stack™
Progress® Chef® Habitat®
Progress® Chef® Infra®
Progress® Chef® Infrastructure Management™
Progress® Chef® Inspec®
Progress® Chef® Premium Content™
Progress® Chef® Supermarket™
Progress® Chef® WorkStation™

DIGITAL DECISIONING

Figure 2: To avoid the community doing something stupid – in Chef's opinion – the manufacturer also added a copyright hammer to its arsenal to back up the license club.

are no longer to distribute the resulting packages under the Chef label. Additionally, third-party vendors must ensure that compiled Chef packages do not share paths on the filesystem with the official packages.

If you want to start your own Chef distribution on the basis of free source code, you have to change the path to, say, `/opt/<Something>`, because the official packages have already claimed `/opt/chef`. The vendor has gracefully allowed the Ruby modules from the official Chef packages to keep their names. However, if you want to switch from Chef to a different Chef distribution, you can look forward to a substantial amount of migration work, if only because of the different paths in the filesystem.

Decisive Action from Debian

With a short delay, distributors started to respond to the confusing path taken by Chef. At Debian GNU/Linux, for example, all of the Chef packages were thrown out of the distribution. A small side note in the release notes is all that remains of Chef in Debian. Anything else was hardly to be expected; after all, the Debian project has enough experience with stubborn vendors. The example of Iceweasel, the Firefox counterpart, remains unforgotten. It was maintained in Debian because Firefox wanted to prevent the browser being distributed under the trademarked name of Firefox (Figure 3).

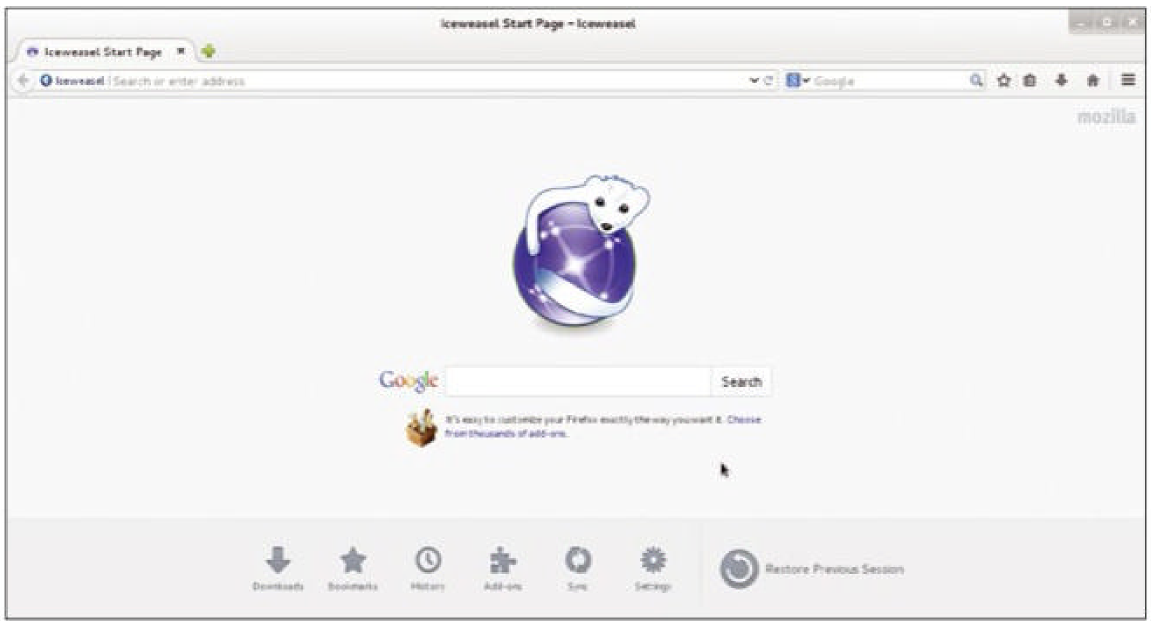
The Debian maintainers apparently do not think that Chef is important enough to warrant providing a separate variant. Therefore, Debian users who used Chef with the distribution thus far have been left out since Debian GNU/Linux 11. You can expect other distributions to follow their example in the mid to long term. The combination of the new Chef license and the obvious implications of the trademark strategy leave them practically no choice.

Alternatives

What are the options for Chef users who have now lost their automator because of the scenario just described? Some may feel tempted at the first moment to fire up a compiler themselves. Where there is no plaintiff, there is no judge – and Progress Software might not even care whether the specially compiled packages use `/opt/chef/` as their as path or something else, as long as you don't redistribute the binaries. This approach only seems attractive at first glance. Anyone who has ever maintained packages of any kind for a Linux distribution will be aware that if you want to build high-quality packages, this strategy will regularly cause overhead on a scale that many companies probably cannot afford.

This is where Cinc [1] enters the scene. Cinc (or CINC) is the acronym for "Cinc is not Chef" and jokingly refers to itself as the "free-as-in-beer distribution" of Chef [2]. Free-as-in-free-beer is a stock phrase in the FLOSS community: Former Free Software Foundation (FSF) boss Richard Stallman regularly used the term to explain which meaning of "free" was meant.

One way or another, Cinc wants to establish itself on the market as a binary-compatible Chef distribution, without incurring costs for admins and avoiding the general wailing and gnashing of teeth provoked by Progress Software. To achieve this goal, the creators behind the project are proceeding in



It's easy to customize your Firefox exactly the way you want it. Choose from thousands of add-ons.

Figure 3: The Chef license changes bring back memories of the Firefox web browser, which had to be renamed Iceweasel in Debian GNU/Linux for quite a while because of bickering over trademark rights.

phases, taking on the various components of Chef one by one.

What Chef Is Made Of

It takes a little detour into Chef's structure to understand in more detail how much progress Cinc has made thus far. Anyone looking at the vendor's portfolio is faced with a large number of components: Chef Automate, Chef Infra, Chef Workstation, Chef InSpec – it's easy to lose track. So, which tool does exactly what in the Chef circus? First, you would do well to focus on Chef Infra, the successor to what admins simply knew as Chef (Figure 4).

Before the automation suite started to add factors such as continuous integration and continuous delivery (CI/CD), the predecessor of today's Chef Infra was basically Chef: It included the Chef server and the client. In combination with cookbooks, these components were all you needed in Chef to automate the provisioning of large fleets with a complete configuration. Chef has always used a server-client model, although local changes on individual systems can also be imported with the Chef components.

Chef Infra comprises three subcomponents. (1) The server is at the core of an automation environment based on Chef. It contains all configuration parameters for all servers in the environment. The cookbooks, which provide the target systems with a specific configuration, are also available here. The Chef server can be queried with a centralized API. It also stores the policy component, which provides a schema for authorizations and processes in Chef. Last but not least, Chef Server is connected to a directory by the Supermarket, from which additional cookbooks or other components can be installed.

The Chef server corresponds to the Chef Infra Client, which runs as an agent on each individual system and receives commands defined by the Chef Server for the respective host so that it can execute them locally. The Chef Workstation, which is itself part of the solution, forms a kind of link between the admin and Chef Infra and is where you will find familiar tools such as Knife, which is used to store the appropriate configurations for each host on the Chef Infra server.

What Cinc Does

Currently, Cinc cannot cover the complete toolset of Chef Infra with its own tools. At the time of writing in early 2022, the Cinc developers had their own versions of Chef Infra Client and a preproduction version of Chef Infra Workstation in their portfolio. In the background, work on a Cinc counterpart to Chef Infra Server was in full swing, so full compatibility with the "real" Chef should be achieved soon.

It's important to remember that building a Chef community distribution that has the official blessing of Progress Software is anything but a trivial and enjoyable experience. In fact, the Cinc people started from scratch with every component. To begin, you need to build packages with customized paths that no longer clash with the official Chef packages. Next, the "Chef" brand has to be removed wherever it could insinuate to the end user that it is an official Progress Software product. Once again, a comparison with Iceweasel in Debian comes to mind. The amount of work the project had to invest in debranding Firefox at the time was huge – especially in light of the fact that for many years it had to be repeated every time a new version was released. The Cinc makers face a similar task. The people behind Cinc vehemently insist that they are not affiliated with Progress Software in any way, and who could blame them?

I need to state explicitly again that Cinc is looking to be as compatible as possible with Chef, to the extent that this is legally permissible. The developers intend for cookbooks designed for a specific Chef version to work with the same version of Cinc. There are limits to compatibility, however, in places such as filesystem paths, precisely because they are required to differ from the paths used by Chef. In this respect, Cinc cannot become a drop-in replacement.

Cinc InSpec Works Well

The counterpart to Chef InSpec [3] is probably the most advanced component in the Cinc portfolio (Figure 5). This component was not part of Chef in the beginning and was acquired by Progress Software.

InSpec stands alongside Chef Infra. Whereas Infra is responsible for configuring the systems, InSpec mainly takes care of monitoring. With the use of InSpec's own declarative scripting language, you store compliance rules for each system. InSpec then automatically checks the systems for compliance. If, for example, the `/etc/passwd` file is not allowed to contain a specific entry, you need to specify this in the configuration and

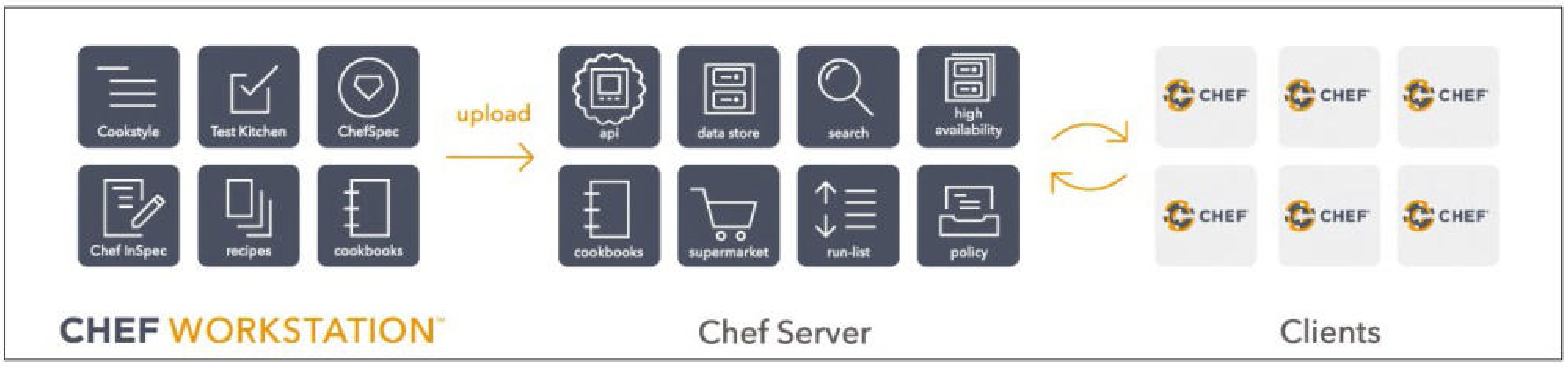
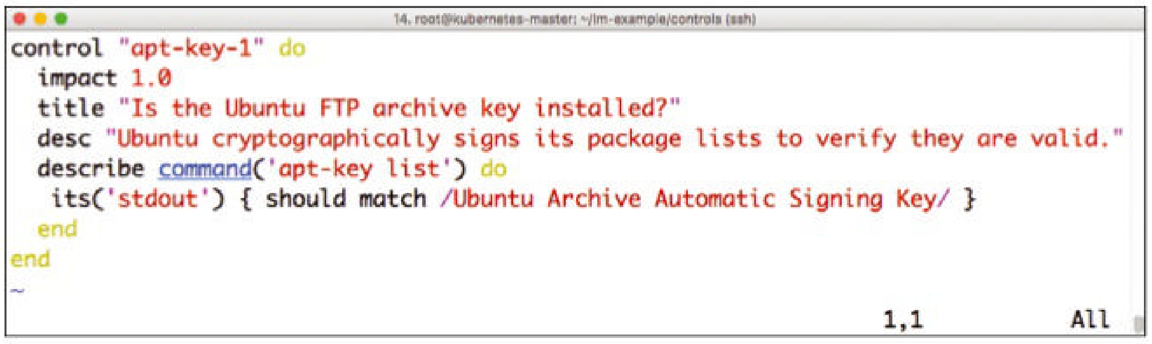


Figure 4: Chef Infra comprises the server, the client, and the workstation. It is joined by the InSpec compliance tester with the Habitat CI/CD tool. © Chef



```
control "apt-key-1" do
  impact 1.0
  title "Is the Ubuntu FTP archive key installed?"
  desc "Ubuntu cryptographically signs its package lists to verify they are valid."
  describe command('apt-key list') do
   its('stdout') { should match /Ubuntu Archive Automatic Signing Key/ }
  end
end
~
                                                                  1,1           All
```

Figure 5: InSpec uses a declarative scripting language used to define conditions. If the conditions are not in place on the system, the tool raises an alert.

then call InSpec on the respective system. If InSpec finds the entry in `passwd`, the alarm bells are set off. Because InSpec is also trademarked, Cinc named its counterpart component Auditor. A quick check with existing definitions for host compliance showed that Cinc Auditor is fully compatible with InSpec. In our lab, the input and output of Chef InSpec and Cinc Auditor were like two peas in a pod. If the Cinc people can bundle the other parts of Chef into their distribution with the same meticulous care, there is no reason for admins not to move to Cinc instead of Chef (Figure 6).

Chef's Distribution Guidelines

By the way, the Chef makers themselves do not think they have gone too far with the license change. In its own FAQ [4], the company states that the Chef code currently does not contain any checks that automatically classify the workload onsite with a view to penalize license violations. The reason given is that the company is aware of its own responsibility for those setups in which Chef is used as a mission-critical component. These setups can themselves be relevant, say, for operating mission-critical infrastructure. Checks of this kind are not planned for the foreseeable future, the statement continues. Whether you actually believe promises like this from the vendor is, of course, a question of your personal stance.

In the same document, the manufacturer does anticipate the option of a fork or an external distribution. However, Chef would not be Chef if it did not also try to jump onto this bandwagon, too. As a precaution, Chef has therefore presented a Distribution Guideline, to which distributors are required to adhere. According to the manufacturer, the respective creators are responsible for the content of distributions and forks.

Also in that document, Progress Software explicitly refuses to provide build help to external distributors from the community. Progress Software states that it is not possible to determine whether a Chef distribution is compatible with the distributor guidelines and the trademark requirements because of a lack of resources for external projects. In plain language: Distributors such as Cinc will discover whether they are compliant at the latest when they receive mail from Chef's legal team.

Conclusions

Cinc could well be a safe haven for desperate admins who don't want to get into trouble because of license violations. Migrating from an older version of Chef to a current Cinc release does involve a few obstacles, especially in terms of the paths to files on the file system. However, a short, sharp shock is probably preferable to infinite pain, and making the move is far less time-consuming than maintaining your own packages – which ultimately makes it acceptable.

What is clearly less acceptable is Progress Software's audacity in forcing admins to resort to these emergency measures. Admittedly, every software vendor is free to determine its business model and change it over time, but if a change results in basically eradicating the technical basis of existing setups, it is tantamount to a severe breach of trust between the vendor and the user. Cinc is bridging the gap right now, which could lead to a rude awakening for Chef. If the Cinc makers manage to keep the quality of their product high, the distribution could attract users away from Chef, who would only have itself to blame. ■

Info

[1] Cinc: [https://cinc.sh/start/]

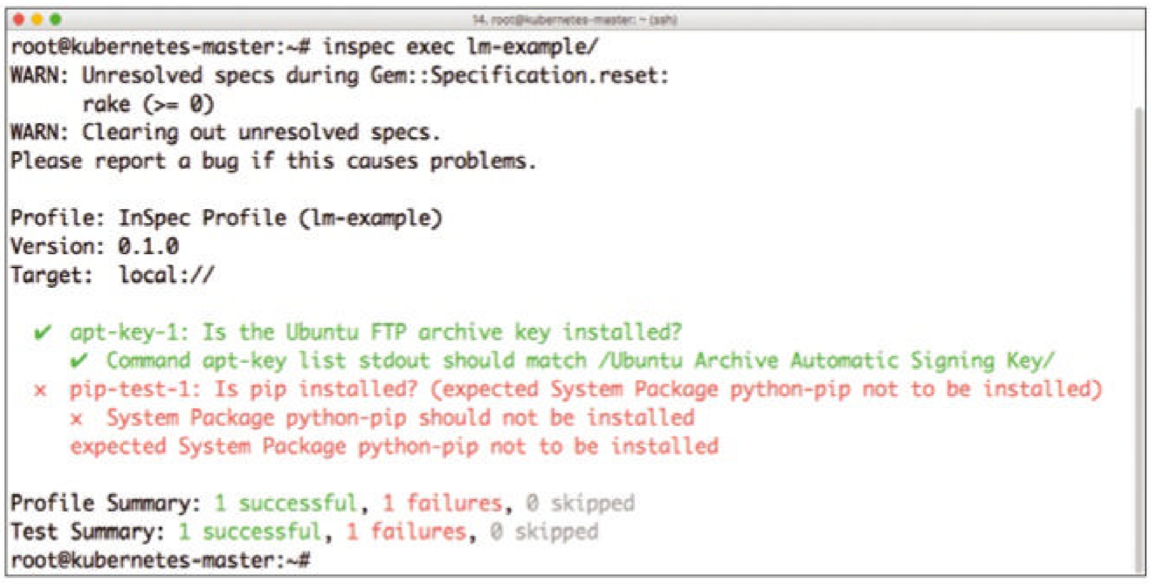
[2] Cooking with Cinc: [https://cinc.sh/blog/cooking_with_cinc/]

[3] "Automated Compliance Testing with InSpec" by Martin Gerhard Loschwitz, *ADMIN*, 2017, issue 42, pg. 64, [https://www.admin-magazine.com/Archive/2017/42/Automated-compliance-testing-with-InSpec/]

[4] Chef FAQ: [https://www.chef.io/pricing/subscription-model-faq]

The Author

Freelance journalist Martin Gerhard Loschwitz focuses primarily on topics such as OpenStack, Kubernetes, and Chef.



```
root@kubernetes-master:~# inspec exec lm-example/
WARN: Unresolved specs during Gem::Specification.reset:
      rake (>= 0)
WARN: Clearing out unresolved specs.
Please report a bug if this causes problems.

Profile: InSpec Profile (lm-example)
Version: 0.1.0
Target:  local://

  ✔  apt-key-1: Is the Ubuntu FTP archive key installed?
     ✔  Command apt-key list stdout should match /Ubuntu Archive Automatic Signing Key/
  ×  pip-test-1: Is pip installed? (expected System Package python-pip not to be installed)
     ×  System Package python-pip should not be installed
     expected System Package python-pip not to be installed

Profile Summary: 1 successful, 1 failures, 0 skipped
Test Summary: 1 successful, 1 failures, 0 skipped
root@kubernetes-master:~#
```

Figure 6: Cinc Auditor, the counterpart to InSpec (pictured here), is already working well and can be used in production.

PowerShell OSDBuilder module

Image Manager

The PowerShell OSDBuilder module is designed to help admins maintain their system images while enforcing security. By Thomas Bär and Frank-Michael Schlede

Many system administrators use images to provide customized Windows installations that comply with company policies and requirements, so images of Windows are an important tool to help them keep their systems up to date while enforcing security. The PowerShell OSDBuilder module is designed to help manage the overhead of maintaining these images.

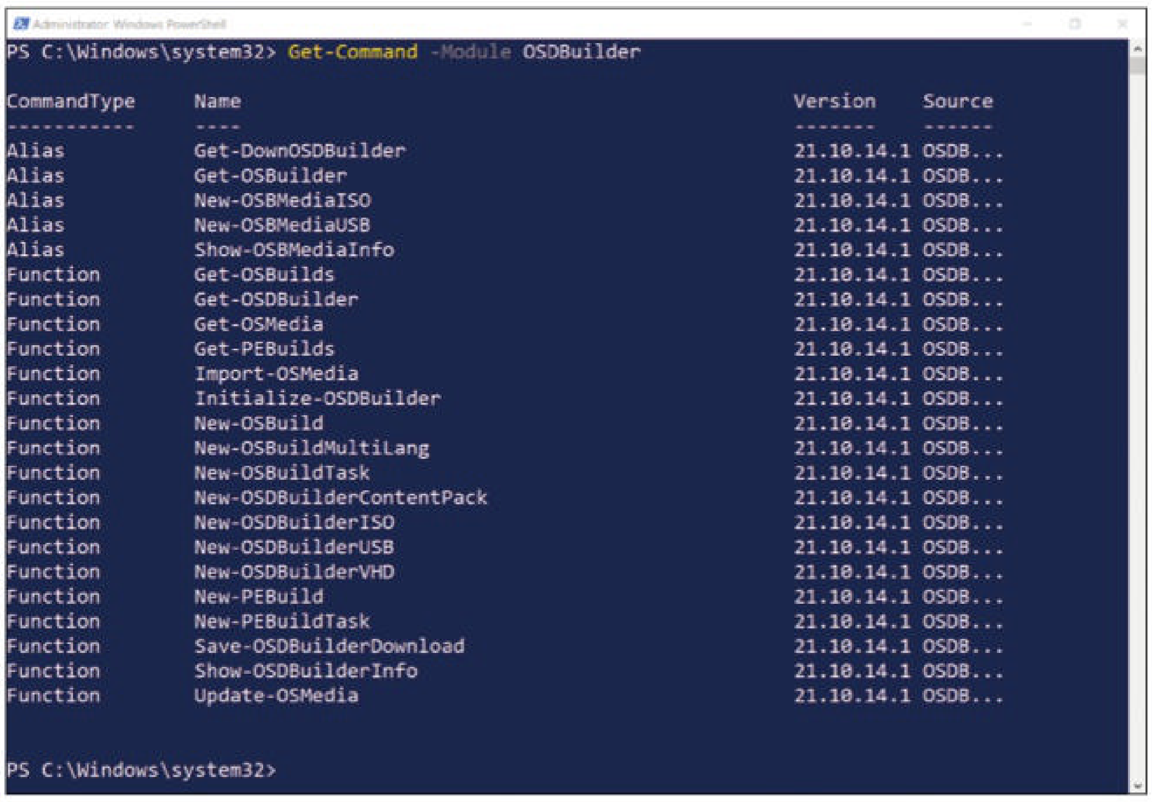
Online vs. Offline

Administrators are always happy to argue about which is the "better way" to manage Windows images: online or offline; however, there is no one answer to this question. If you want to maintain your Windows images online, you first need to set up a Windows installation (usually on a virtual machine) that suits your needs and then grab an image of it, either with Microsoft tools or third-party software such as Baramundi. This image then needs to be started up again for retroactive updates and refreshed after applying the updates. One argument in favor of the offline method, on the other hand, is the possibility of including updates automatically to the Windows images. One disadvantage of this method is that it is not possible to include desktop applications in a Windows image.

Neither method delivers an unequivocal knock-out factor. The IT department needs to decide which approach is the best fit for their systems. The PowerShell OSDBuilder module (Figure 1) offers one potential approach to managing images offline, which you then distribute with System Center Configuration Manager (SCCM) or some other tool. One big advantage of OSDBuilder is that all information is stored in a response file, so the file contains everything you need to update the image in question.

Installing the OSDBuilder Module

We used a newish version of Windows 10 21H1 (build 19043.1320) running on a virtual machine on VMware Workstation 16 for this example. The PowerShell commands were tested in both version 5.1, which is found by default on Windows 10, and the current version 7.2; they have only a few differences, and we will point them out at the appropriate points in the article. Installing OSDBuilder is like installing



```
PS C:\Windows\system32> Get-Command -Module OSDBuilder

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           Get-DownOSDBuilder                                 21.10.14.1 OSDB...
Alias           Get-OSBuilder                                      21.10.14.1 OSDB...
Alias           New-OSBMediaISO                                    21.10.14.1 OSDB...
Alias           New-OSBMediaUSB                                    21.10.14.1 OSDB...
Alias           Show-OSBMediaInfo                                  21.10.14.1 OSDB...
Function        Get-OSBuilds                                       21.10.14.1 OSDB...
Function        Get-OSDBuilder                                     21.10.14.1 OSDB...
Function        Get-OSMedia                                        21.10.14.1 OSDB...
Function        Get-PEBuilds                                       21.10.14.1 OSDB...
Function        Import-OSMedia                                     21.10.14.1 OSDB...
Function        Initialize-OSDBuilder                              21.10.14.1 OSDB...
Function        New-OSBuild                                        21.10.14.1 OSDB...
Function        New-OSBuildMultiLang                               21.10.14.1 OSDB...
Function        New-OSBuildTask                                    21.10.14.1 OSDB...
Function        New-OSDBuilderContentPack                          21.10.14.1 OSDB...
Function        New-OSDBuilderISO                                  21.10.14.1 OSDB...
Function        New-OSDBuilderUSB                                  21.10.14.1 OSDB...
Function        New-OSDBuilderVHD                                  21.10.14.1 OSDB...
Function        New-PEBuild                                        21.10.14.1 OSDB...
Function        New-PEBuildTask                                    21.10.14.1 OSDB...
Function        Save-OSDBuilderDownload                            21.10.14.1 OSDB...
Function        Show-OSDBuilderInfo                                21.10.14.1 OSDB...
Function        Update-OSMedia                                     21.10.14.1 OSDB...


PS C:\Windows\system32>
```

Figure 1: The OSDBuilder module adds new cmdlets, displayed here with `Get-Command`.

Photo by Shubham Dhage on Unsplash

any other module in PowerShell; just type the command:

```
Install module -Name OSDBuilder
```

The module must be installed from the PowerShell Gallery (PSGallery) repository [1]. Windows classifies this repository as untrusted by default. PowerShell version 5 and version 7 both draw your attention to this fact after you enter the commands for installation, and you are prompted to decide whether you really want to do this. Installation is possible immediately after confirming in version 7, whereas you need to download and install the matching NuGet manager first in version 5. NuGet is a package manager for .NET that helps developers create, share, and use .NET libraries. After completing this step, you can then install the OSDBuilder module. To avoid problems when executing scripts, the recommendation is to lift this restriction by running the next command on the system in question:

```
Set-Executionpolicy Bypass -force
```

The `-force` parameter is not absolutely necessary here, but it effectively prevents additional prompts from appearing.
Now download a current ISO file for Windows 10 21H1, version dated October 2021, from your Visual Studio account with Microsoft and double-click to mount it as drive `F:` on the virtualized system. Our test system has a partition size of approximately 100GB that is mapped as drive `E:`. We will be using this for OSDBuilder, which is why we need to assign the corresponding path to the module in the next step:

```
Get-OSDBuilder -SetPath E:\OSDBuilder
```

This command also specifies the target folder for the images to be created. The module now displays a mass of information on the screen. You will discover that the version of OSDBuilder you are using works with Windows 10 starting in version 1607 up to and including version 21H1 and that it supports Windows 11 version 21H2 and Windows Server starting with version 2016 1607 up to version 2022 21H1. Additionally, some shortcuts for further processing are listed, including the call

```
Get-OSDBuilder -CreatePaths
```

which creates a directory structure (Figure 2) to suit the previously specified path. Now that all the prep work is complete, it is time to select the medium to be installed.

Importing and Maintaining the Image

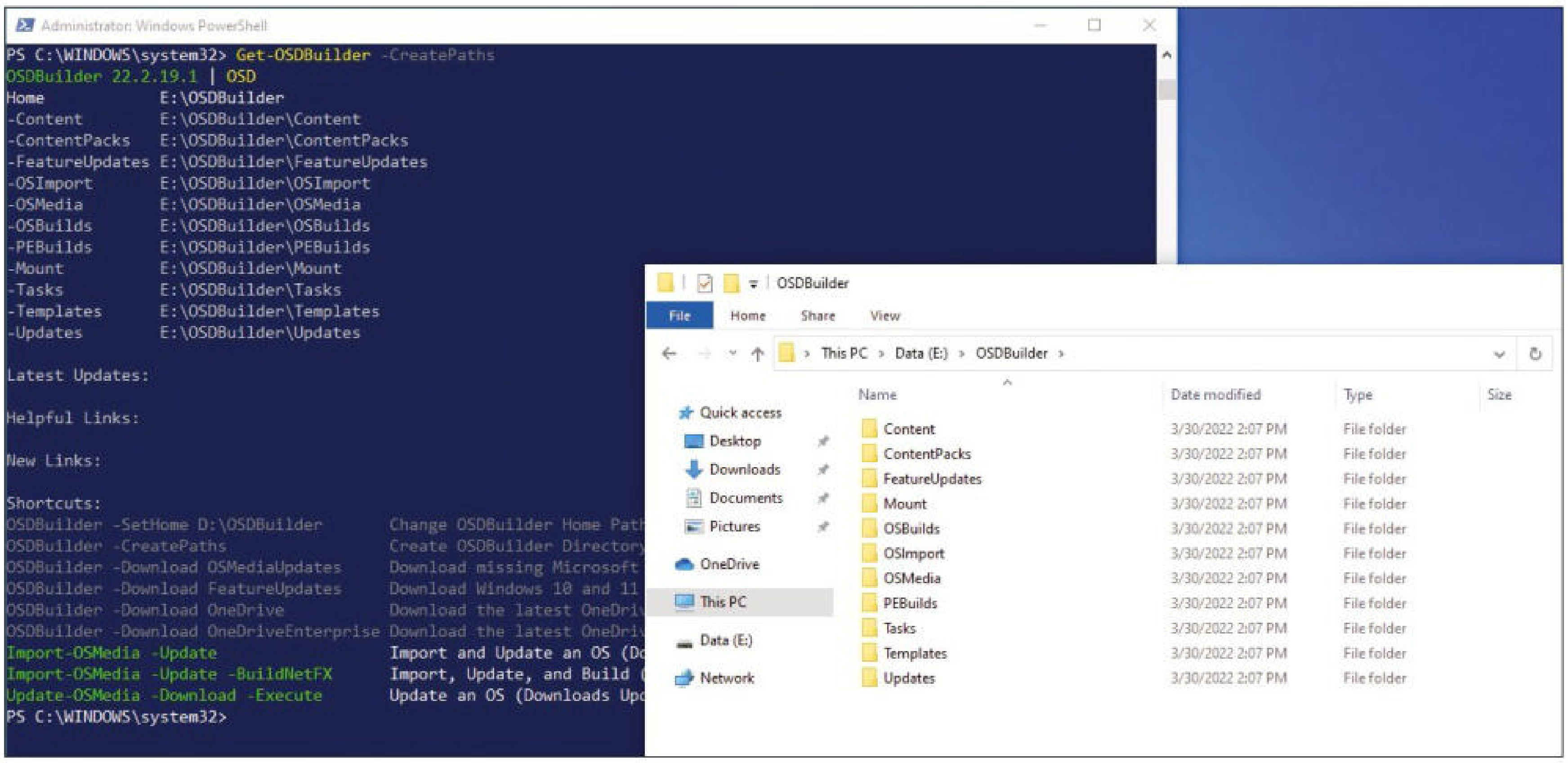
The DVD images for Windows officially provided by Microsoft usually contain a whole series of images with the different editions of the operating system. Because the image was already mounted as a virtual drive on the filesystem, the command

```
Import-OSMedia
```

at the PowerShell command line (Figure 3) scans the mounted DVD and lists its content; it also works if you have several disks mounted on your system. OSDBuilder now displays the entire content of the different DVDs, and you can select the desired edition in the listing window. We used Windows 10 Pro (image index number 5) for this test. Next, you need to import the corresponding media into the designated directories. In the process, OSDBuilder outputs information about the imported Windows image (WIM) index while creating a text logfile in the `logs` directory. The next step is to update the selected Windows image:

```
Update-OSmedia -Download -Execute
```

You are then presented the selected image once again in a list, where you need to select it; the system then



```
PS C:\WINDOWS\system32> Get-OSDBuilder -CreatePaths
OSDBuilder 22.2.19.1 | OSD
Home            E:\OSDBuilder
-Content        E:\OSDBuilder\Content
-ContentPacks   E:\OSDBuilder\ContentPacks
-FeatureUpdates E:\OSDBuilder\FeatureUpdates
-OSImport       E:\OSDBuilder\OSImport
-OSMedia        E:\OSDBuilder\OSMedia
-OSBuilds       E:\OSDBuilder\OSBuilds
-PEBuilds       E:\OSDBuilder\PEBuilds
-Mount          E:\OSDBuilder\Mount
-Tasks          E:\OSDBuilder\Tasks
-Templates      E:\OSDBuilder\Templates
-Updates        E:\OSDBuilder\Updates

Latest Updates:

Helpful Links:

New Links:

Shortcuts:
OSDBuilder -SetHome D:\OSDBuilder          Change OSDBuilder Home Path
OSDBuilder -CreatePaths                    Create OSDBuilder Directory
OSDBuilder -Download OSMediaUpdates        Download missing Microsoft
OSDBuilder -Download FeatureUpdates        Download Windows 10 and 11
OSDBuilder -Download OneDrive              Download the latest OneDri
OSDBuilder -Download OneDriveEnterprise    Download the latest OneDri
Import-OSMedia -Update                     Import and Update an OS (Do
Import-OSMedia -Update -BuildNetFX         Import, Update, and Build (
Update-OSMedia -Download -Execute          Update an OS (Downloads Upd
PS C:\WINDOWS\system32>
```

Name	Date modified	Type	Size
Content	3/30/2022 2:07 PM	File folder	
ContentPacks	3/30/2022 2:07 PM	File folder	
FeatureUpdates	3/30/2022 2:07 PM	File folder	
Mount	3/30/2022 2:07 PM	File folder	
OSBuilds	3/30/2022 2:07 PM	File folder	
OSImport	3/30/2022 2:07 PM	File folder	
OSMedia	3/30/2022 2:07 PM	File folder	
PEBuilds	3/30/2022 2:07 PM	File folder	
Tasks	3/30/2022 2:07 PM	File folder	
Templates	3/30/2022 2:07 PM	File folder	
Updates	3/30/2022 2:07 PM	File folder	

Figure 2: Using PowerShell to create the directory structure in the target directory.

starts downloading the updates and integrating them into the image. Finally, take a quick look at the tasks mentioned at the beginning. You can use these commands to initiate a build process and create media with your choice of updates:

```
New-OSBuildTask
    -TaskName <name of task>
    -EnableFeature <Feature to be enabled>
```

An example could look like:

```
New-OSBuildTask ➘
  -TaskName "Install NetFx" ➘
  -EnableFeature -EnableNetFx3
```

To run the build process for the selected image at the same time, change the command to:

```
New-OSBuild -Download -Execute ➘
  -EnableNetFX -SelectUpdates
```

Pressing the *-Download* button downloads any missing updates directly from Microsoft and applies them to the image. The *-Execute* parameter triggers the build process directly, whereas *-SelectUpdates* brings up a window where you can choose from the available updates. If you run the call this way, the module prompts you for the name of the matching task. If you want this update to be "taskless," you need the command:

```
New-OSBuild -Download -Execute ➘
  -EnableNetFX -SelectUpdates -Skiptask
```

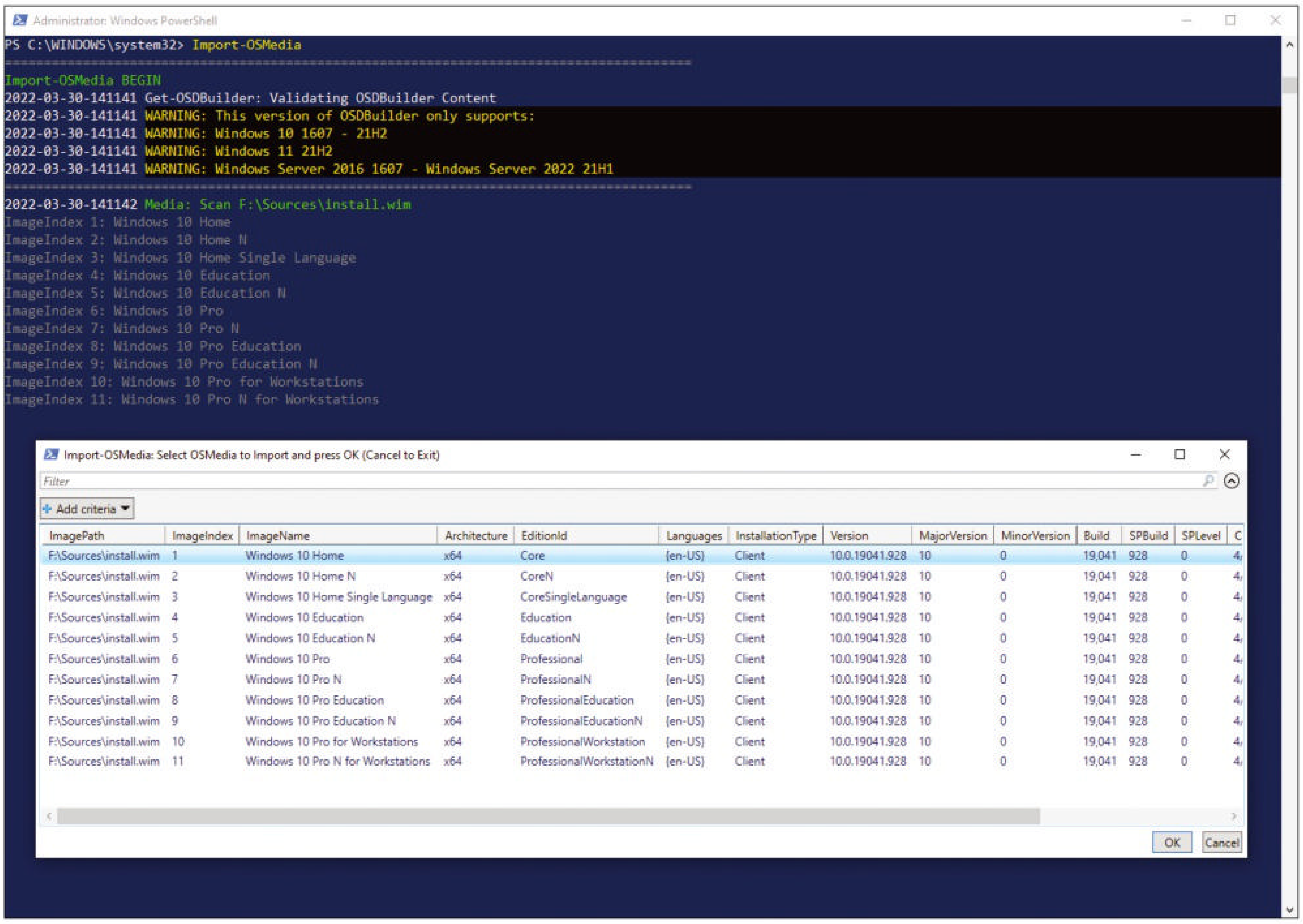
After selecting the imported OS image, the module again provides a grid view with the updates from which to choose, including the service.

Conclusions

Even if you have not previously looked into the subject of maintaining Windows images offline, you are likely to make the acquaintance of the PowerShell OSDBuilder module when confronted with this task. Not only is this module simple to install and easy to use, it also works smoothly and without any major differences between PowerShell versions 5 and 7. What we particularly liked is the way the module takes the user by the hand and guides them through the installation and upgrade processes. Moreover, because OSDBuilder logs all steps in great detail and makes it easy to automate offline image maintenance with the help of tasks, the process is very practical. Anyone who has Windows images they need to look after and maintain – and this is probably going to be the case in most IT organizations – will definitely want to take a look at this PowerShell module. Of course, it does not handle software distribution, but it does complement software distribution tools in a useful way. ■

Info

[1] PSGallery: [https://www.powershellgallery.com]

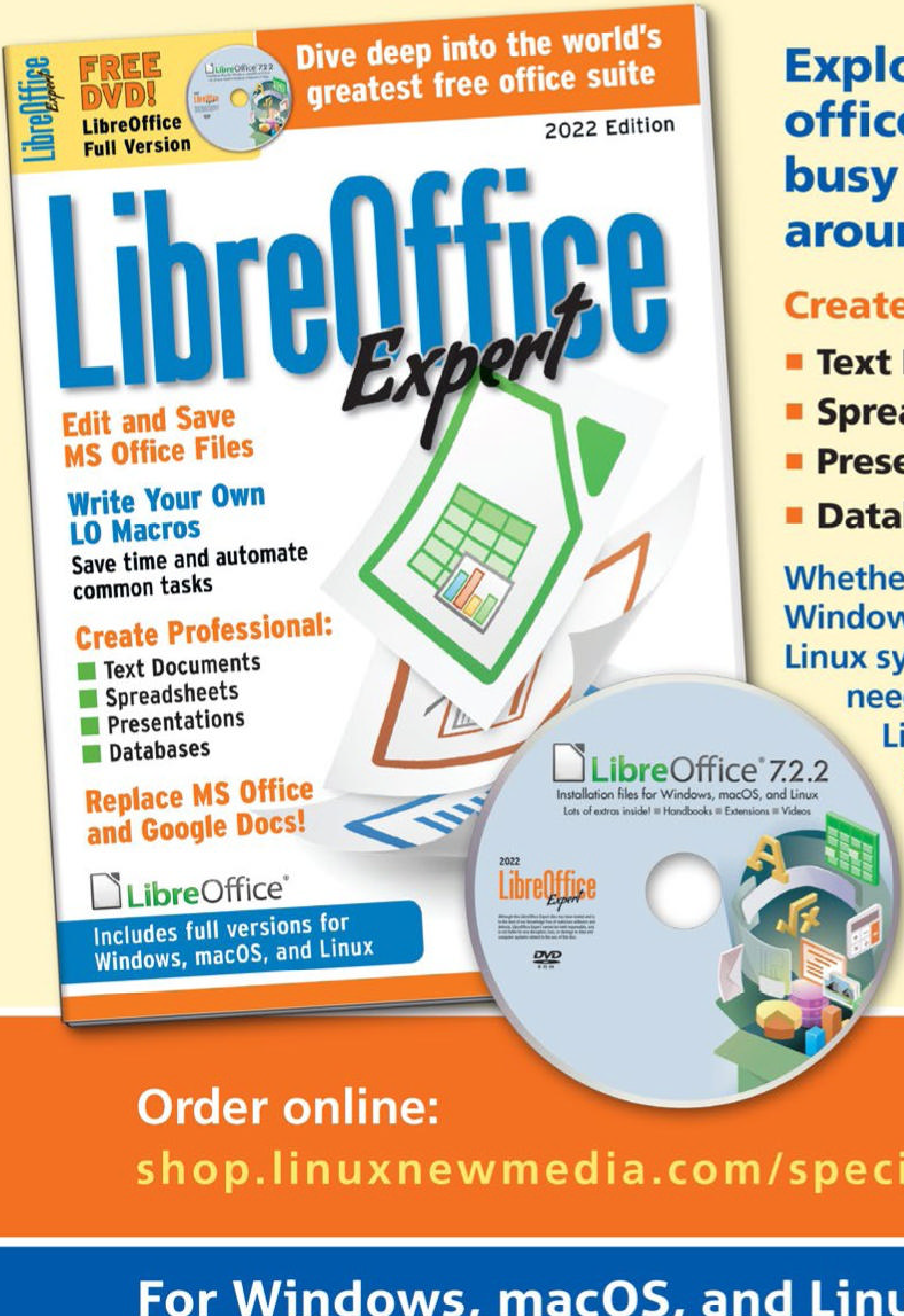


```
PS C:\WINDOWS\system32> Import-OSMedia
========================================================================================
Import-OSMedia BEGIN
2022-03-30-141141 Get-OSDBuilder: Validating OSDBuilder Content
2022-03-30-141141 WARNING: This version of OSDBuilder only supports:
2022-03-30-141141 WARNING: Windows 10 1607 - 21H2
2022-03-30-141141 WARNING: Windows 11 21H2
2022-03-30-141141 WARNING: Windows Server 2016 1607 - Windows Server 2022 21H1
========================================================================================
2022-03-30-141142 Media: Scan F:\Sources\install.wim
ImageIndex 1: Windows 10 Home
ImageIndex 2: Windows 10 Home N
ImageIndex 3: Windows 10 Home Single Language
ImageIndex 4: Windows 10 Education
ImageIndex 5: Windows 10 Education N
ImageIndex 6: Windows 10 Pro
ImageIndex 7: Windows 10 Pro N
ImageIndex 8: Windows 10 Pro Education
ImageIndex 9: Windows 10 Pro Education N
ImageIndex 10: Windows 10 Pro for Workstations
ImageIndex 11: Windows 10 Pro N for Workstations
```

Import-OSMedia: Select OSMedia to Import and press OK (Cancel to Exit)
Filter
Add criteria
OK
Cancel

ImagePath	ImageIndex	ImageName	Architecture	EditionId	Languages	InstallationType	Version	MajorVersion	MinorVersion	Build	SPBuild	SPLevel	C
F:\Sources\install.wim	1	Windows 10 Home	x64	Core	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	2	Windows 10 Home N	x64	CoreN	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	3	Windows 10 Home Single Language	x64	CoreSingleLanguage	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	4	Windows 10 Education	x64	Education	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	5	Windows 10 Education N	x64	EducationN	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	6	Windows 10 Pro	x64	Professional	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	7	Windows 10 Pro N	x64	ProfessionalN	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	8	Windows 10 Pro Education	x64	ProfessionalEducation	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	9	Windows 10 Pro Education N	x64	ProfessionalEducationN	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	10	Windows 10 Pro for Workstations	x64	ProfessionalWorkstation	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,
F:\Sources\install.wim	11	Windows 10 Pro N for Workstations	x64	ProfessionalWorkstationN	{en-US}	Client	10.0.19041.928	10	0	19,041	928	0	4,

Figure 3: If more than one Windows edition is found on the source DVD, OSDBuilder displays all of them and gives you a choice.


FREE DVD!
LibreOffice Full Version
Dive deep into the world's greatest free office suite
2022 Edition
LibreOffice Expert
Edit and Save MS Office Files
Write Your Own LO Macros
Save time and automate common tasks
Create Professional:
Text Documents
Spreadsheets
Presentations
Databases
Replace MS Office and Google Docs!
LibreOffice
Includes full versions for Windows, macOS, and Linux
LibreOffice 7.2.2
Installation files for Windows, macOS, and Linux
Lots of extras inside! Handbooks Extensions Videos
2022 LibreOffice Expert
DVD

Monitoring in the Google Cloud Platform

Cloud Gazer

We introduce monitoring in the Google Cloud Platform by monitoring virtual machines, setting up alerts, observing important metrics in dashboards, and defining service-level objectives. By Guido Söldner

When you are responsible for infrastructure and applications, monitoring is a must-have for gaining insights into the status of the components involved, not only for on-premises environments but also for the public cloud. The Google Cloud Platform (GCP) discussed in this article, along with AWS and Microsoft Azure, is one of the three major public clouds.

In recent years, Google has seen considerable growth, especially in the areas of machine learning and big data. However, GCP is also very much in the running in the classic Internet-as-a-Service (IaaS) arena. Monitoring is definitive for the admin's ability to operate a cloud infrastructure effectively. At Google, the Cloud Operations Suite, which addresses the topics of monitoring, logging, tracing, debugging, profiling, and auditing, was named Stackdriver until two years ago. Google no longer uses this name and simply refers to it as the Operations Suite.

To ensure that monitoring works during operations, you need to field the data from the source systems in the form of signals. In the monitoring world, data is equivalent to metrics, which can come from IaaS components such as virtual machines, from added-value services such as managed databases, from platforms such as Kubernetes, and from microservices – but also from the applications themselves. Keeping track of incidents is essential, whether in the form of alerts, error reports, or even service-level objectives. The Operations Suite lets you consolidate and view the data in more detail, visualize the results, and use them for troubleshooting.

Access to monitoring data on GCP can be both centralized and decentralized. In the Google Cloud, all resources are assigned to projects. In Operations Suite, each project can collect and analyze data on its own. However, if you want to keep track of all systems across the entire organization, you need to add multiple projects to a monitoring workspace.

Virtual Machine Monitoring

When logged into GCP, first switch to the monitoring tool by typing *Monitoring* top center in the search bar and then selecting the first item in the results bar. Alternatively, click directly in the menubar on the left on *Operations* | *Monitoring* | *Overview*. You can then select additional projects top left in the Metrics scope and add them to a workspace.

Another way to view monitoring data is to select the resource directly in the GCP GUI. For virtual machines, simply click on the individual VM and then switch to the *Observability* tab (Figure 1), which will take you to a dashboard that displays the most important metrics. For VMs, these are CPU utilization, various network traffic values such as the number of packets received and sent, firewall blocked packets, and hard disk metrics such as input/output operations per second (IOPS) or data throughput.

Additionally, you can view even more extensive metrics on memory and disk utilization; however, you need to install the Ops Agent on the VMs. Fortunately, the GUI also immediately shows the installation instructions for this step and prompts you to execute the commands shown in Listing 1.

By the way, the Ops Agent is based on the OpenTelemetry standard, a project by the Cloud Native Computing Foundation. Because it does not use a proprietary standard, the Ops Agent will work with all monitoring tools that can handle the standard.

Lead Image © Maksym Chornii, 123rf.com

Uptime Checks

Besides querying basic metrics, defining uptime checks is one of the basic tasks in monitoring. A classic method of implementation is to call an HTTP endpoint and check the HTTP response status. In the case of a virtual machine, you can easily try this implementation with an Apache web server. For example, if you are using Debian, first install the Apache server:

```
sudo apt-get update
sudo apt-get install apache2 php7.0
sudo service apache2 restart
```

The uptime check is then configured in the Operations Suite by selecting *Operations* | *Monitoring* | *Uptime Checks* from the menu on the left and then clicking on + *CREATE UPTIME CHECK* to start the configuration. On the left side of the screen, you will then see the wizard. The first step is simple: Enter a name for the uptime check and then click *Next*. Now you need to specify the target to be checked. Besides URLs, you can also select Kubernetes load balancers, App Engine instances, VM instances, and AWS Elastic load balancers. In this case, the configuration is:

```
- Protocol: HTTP
- Instance: <name of VM>
- Check Frequency: 1 minute
```

At this point, do not forget that you need to set up appropriate firewall rules for the VM to make sure port 80 is also accessible. Leave the other configuration values as they are and go to the next step. In the *Response Validation* menu item, keep the value of 10 seconds for *Response Timeout* and check the *Log check failures* box to enable logging in the event of an error.

Click *Next* one more time to get to the last step in the wizard: configuring alerts and notifications. Leave the *Create an alert* checkbox enabled and change the name of the alert, if necessary. Also leave the *Notification Channel* empty, but if you want, you can send alerts there. Google supports mobile devices such as smartphones, PagerDuty Services, PagerDuty Sync, Slack, webhooks, email, SMS, and the pub/sub internal messaging service. After completing the wizard, it takes a few minutes for the first data to show up in the Operations Suite.

Setting Alerts

Operations Suite also lets you define alerts linked to conditions and can send notifications if required. In GCP, you set up these alerts by selecting the *Operations* | *Monitoring* | *Alerting* menu item in the navigation bar. You can start creating an alert by pressing + *CREATE POLICY* at the top.

To begin, define a condition and press *ADD CONDITION*. A window opens in which you need to select a name for the condition and a corresponding metric. For this scenario, I specified

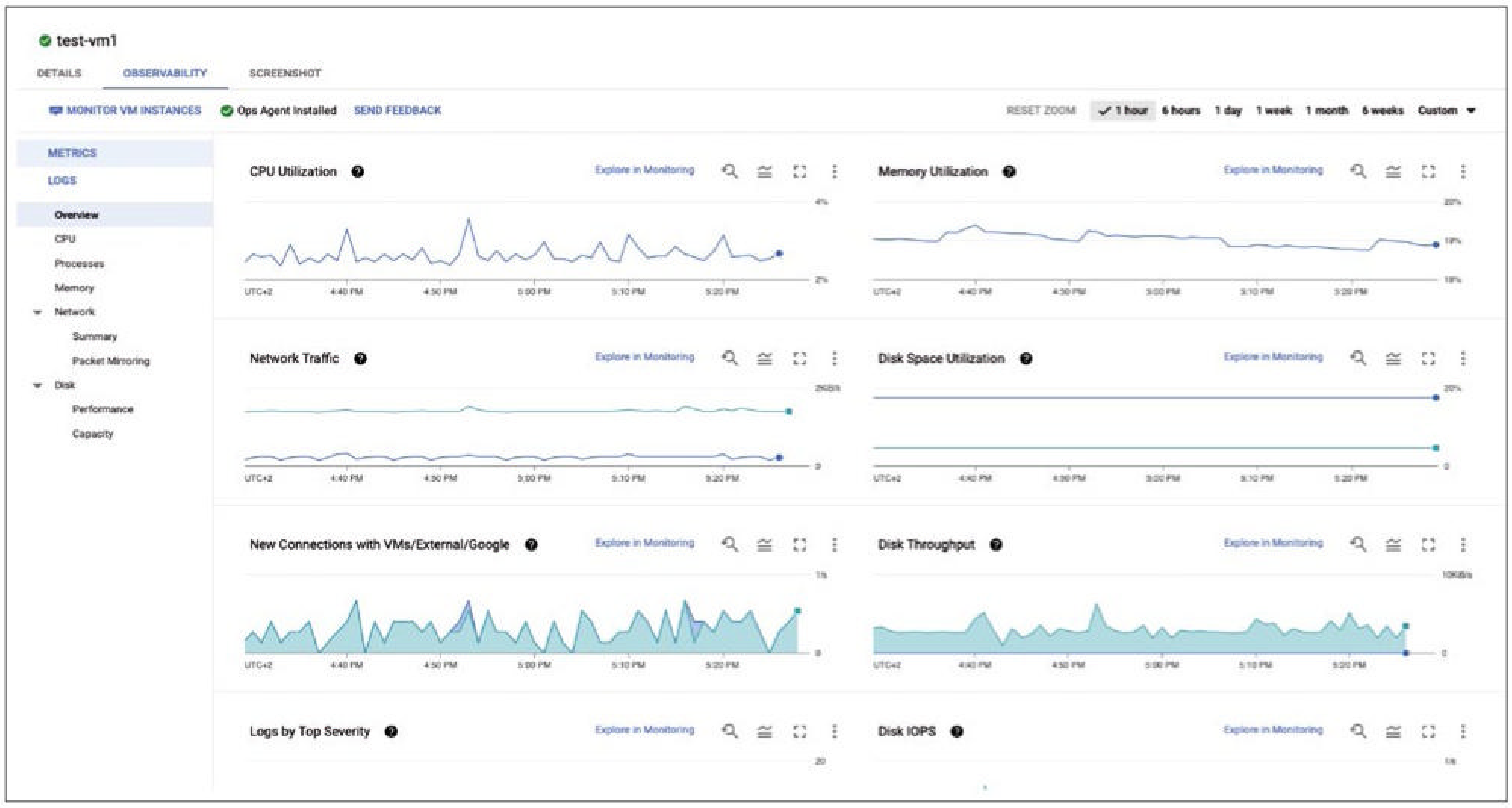


Figure 1: For virtual machines, the ***Observability*** dashboard clearly displays numerous metrics.

Listing 1: Installing the Ops Agent

```
:> agents_to_install.csv &&
  echo '"projects/playground-gs/zones/europe-west3-c/instances/test-vm1","[{""type"":""ops-agent""}]"' > agents_to_install.csv &&
  curl -sSO https://dl.google.com/cloudagents/mass-provision-google-cloud-ops-agents.py &&
  python3 mass-provision-google-cloud-ops-agents.py --file agents_to_install.csv
```

VM Instance as the resource type and network traffic as the metric. Typing *Network* brings up a list of possible metrics, from which I selected *agent.googleapis.com/interface/traffic*. Next, I set the following configuration values,

```
- Condition: is above
- Threshold: 500
- For: 1 minute
```

then clicked *Add* and *Next*. You can now configure the notifications. In the Notification Channels section, select *Manage Notification Channels* and enter your email address in the window that opens. After doing so, you can select the defined email address in the original window and click *OK*.

Clicking *Next* takes you to the last step, where you enter an alert name (e.g., *Inbound Traffic Alert*). Optionally, you can also enter text that you want to add to the email. Clicking *Save* completes the process. After a few minutes, an alert should appear, which you can display on the dashboard.

Creating Dashboards

The Google Cloud Platform already gives you a large number of dashboards for monitoring, but you might still want to create your own. To do so, go to the *Operations* | *Monitoring* | *Dashboards* page, which shows all existing dashboards. Moreover, you can use as inspiration sample templates from a large library on the *SAMPLE LIBRARY* tab.

To create your own dashboard, press *Create Dashboard* and assign a name. Now select elements you want to show on the dashboard from the list of charts. The selection list is quite extensive. In addition to line diagrams, you can choose from stack and bar charts, heat maps, tables, gage charts, scorecards, text, and warnings.

For example, you could start by adding a line diagram as an element and titling it *CPU load*, selecting *VM Instance* as the resource type, and setting the metric to *CPU load (1m)*. Now you can add a second chart by clicking *ADD CHART* at the top (Figure 2). Again, choose a line diagram and specify *Received packets* as the label. The resource type is again *VM Instance*, and the metric this time is *Received packets (gce instance)*.

Monitoring Services

Classic monitoring often focuses on measured values at the resource level. In this case, it was information such as CPU utilization, network packets transmitted, or memory utilization. However, modern applications consist of a large set of individual components, so it is not very useful to look at a very large set of metrics individually. It makes more sense to look at the application as a whole and, in particular, measure the application's most important values – availability and latency.

To explain how this works, I'll look at a sample application that is publicly available on GitHub and is based on Google's App Engine service. To deploy it, open the cloud shell in the GCP console and clone the repository as follows: First choose the region that suits your use case (I chose Western Europe),

```
git clone https://github.com/haggman/↩
  HelloLoggingNodeJS.git
```

Next, deploy the application with the commands

```
cd HelloLoggingNodeJS
gcloud app create --region=europe-west3
gcloud app deploy
```

The deployment process takes one or two minutes, and after it completes, you will see the URL of the application in the shell output. On opening the web page, no big surprises jump out: It's yet another "Hello World!" Now the task is to generate some load on the application with the code snippet

```
while true; ↩
  do curl -s https://$DEVSHELL_PROJECT_ID.↩
             appspot.com/random-error ↩
          -w '\n' ;sleep .1s; done
```

The cloud shell now continuously generates log output, which you can safely ignore. To monitor the service, you need to familiarize yourself with the definitions of service-level indicators (SLIs) and service-level objectives (SLOs). SLI metrics measure the reliability of a service. To do this, select a metric, divide the positive events by the number of total events, and multiply by 100 – for example, SLI (%) = (Positive events/Valid events) x 100.

The classic SLI values are availability or latency, of which you do not want to exceed a certain limit. You can define SLOs on this basis as an agreement of compliance with certain metric values. SLOs need to be measurable metrics that have been

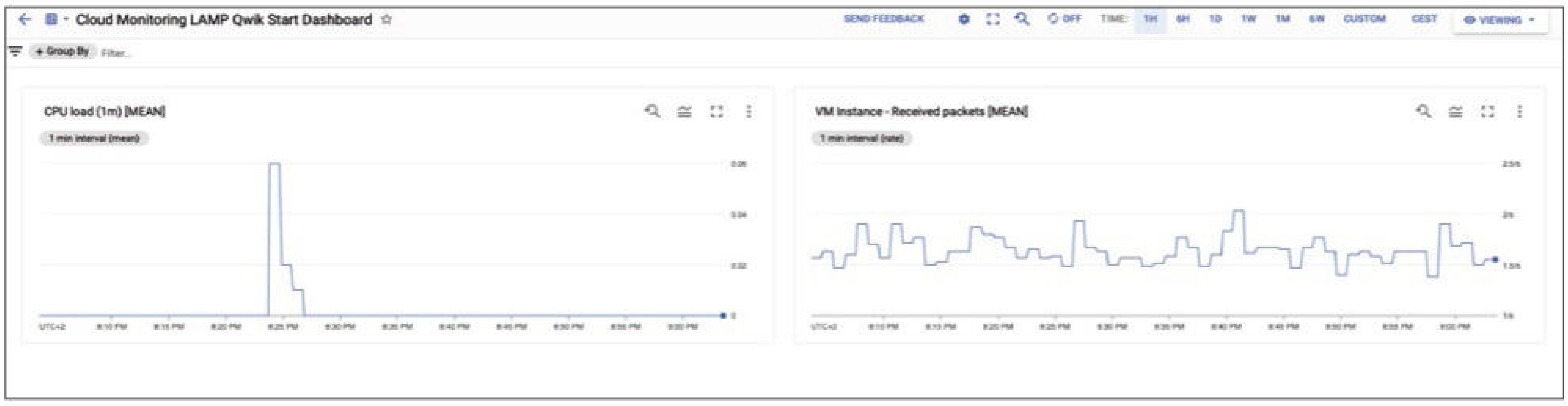


Figure 2: **Creating your own charts in GCP usually takes just a few minutes.**

documented and shared among the stakeholders. SLOs are also often part of a service-level agreement (SLA) that documents the key performance indicators (KPIs) the customer expects from a provider.

Creating SLOs

In the App Engine application [1] cloned earlier in this article, approximately every 1,000th call throws an error. To set up service monitoring, go to the left-hand menu in the GPC GUI and select *Operations* | *Monitoring* | *Services*. You will immediately notice that the App Engine application is already displayed on the dashboard (Figure 3). Click on the *Default* link to view details about the service and start creating an SLO by pressing + *CREATE SLO* on the right side. Again, this opens a wizard:

- In the first step, select *Availabiliy* as the metric. Leave the *Request-based* checkbox checked at the bottom, which means that availability is calculated over the entire period, regardless of the load. Pressing *Continue* takes you to the next page.
- Now check the SLI settings. If you wait awhile, you will see a preview of the requests.
- On the third page, configure the SLO, setting the *Period type* to *Rolling*, the *Period length* to seven days, and a *Performance goal* of 99.5 percent. Clicking *Continue* takes you to the next step in the wizard.
- The last page gives you an overview with the option of viewing the configuration you created in JSON. Choosing *CREATE SLO* closes the wizard.

When you look at the SLO, everything seems to be in the green zone. The service-level indicator, the error budget, and the alerts are as expected. The concept of the error budget comes from Google, describing a measurable number of errors or a percentage of the time a service might not be available and still be considered a valid SLO. In this example, this value is 100 percent minus 99.5 percent, or 0.5 percent. A product team can use the current error budget to carry out maintenance work, for example. When the budget is exhausted, the work just has to wait until the error budget has replenished.

Regardless of the values, you might want to send warnings if an SLO is not reached. To do this, click the *CREATE SLO ALERT* link on the SLO. You again need to configure a number of parameters:

- *Lookback duration* determines the period of time over which you

want to look at data. Longer values are especially interesting for compliance, and shorter values give you a quicker warning if errors occur. For this example, I'll just go for *10* (hours).

- *Burn rate threshold* determines how fast the error budget might be used up. A value of *1* on a period indicates that the error budget is used up exactly in a period. I configured *1.5* here.

The next steps regarding the Notification Channel should be familiar, and you can go on to complete the wizard.

If you want to provoke more errors, you can modify the application accordingly. If you open the `index.js` files in the cloud shell and search for *random-error*, you can massively increase the number of errors in this line by reducing the number *1,000* to a lower value (e.g., *20*). You can then use the

```
gcloud app deploy
```

command to redeploy the application. As soon as you call the application again, the SLO count should decrease rapidly.

Network Monitoring with VPC Flow Logs

Besides monitoring individual resources, network monitoring is a fundamental task in IT operations. Google VPC gives you flow logs for this that can be enabled on any subnet to log information about transmitted and received network packets and is pure log data for now. Nevertheless, it can be interesting for many use cases to create your own metrics on the basis of these logs (e.g., how often packets were dropped on a network).

VPC Flow Logs are easily enabled by switching to *Networking* | *VPC network* | *VPC networks* in the navigation menu on the left and clicking on the subnet to be configured. When you get there, first switch to Edit mode at the top and then enable the flow logs at the bottom. Once you select *On*, you can set the aggregation period as well as the sample rate. For most scenarios, the default settings of *5* seconds for aggregation and the sample rate of *50* will be good choices. The latter defines how many values are forwarded to logs. What is also interesting is the estimate below this of how many megabytes per day are aggregated for the subnet. The individual data itself can be viewed in the Logging section.

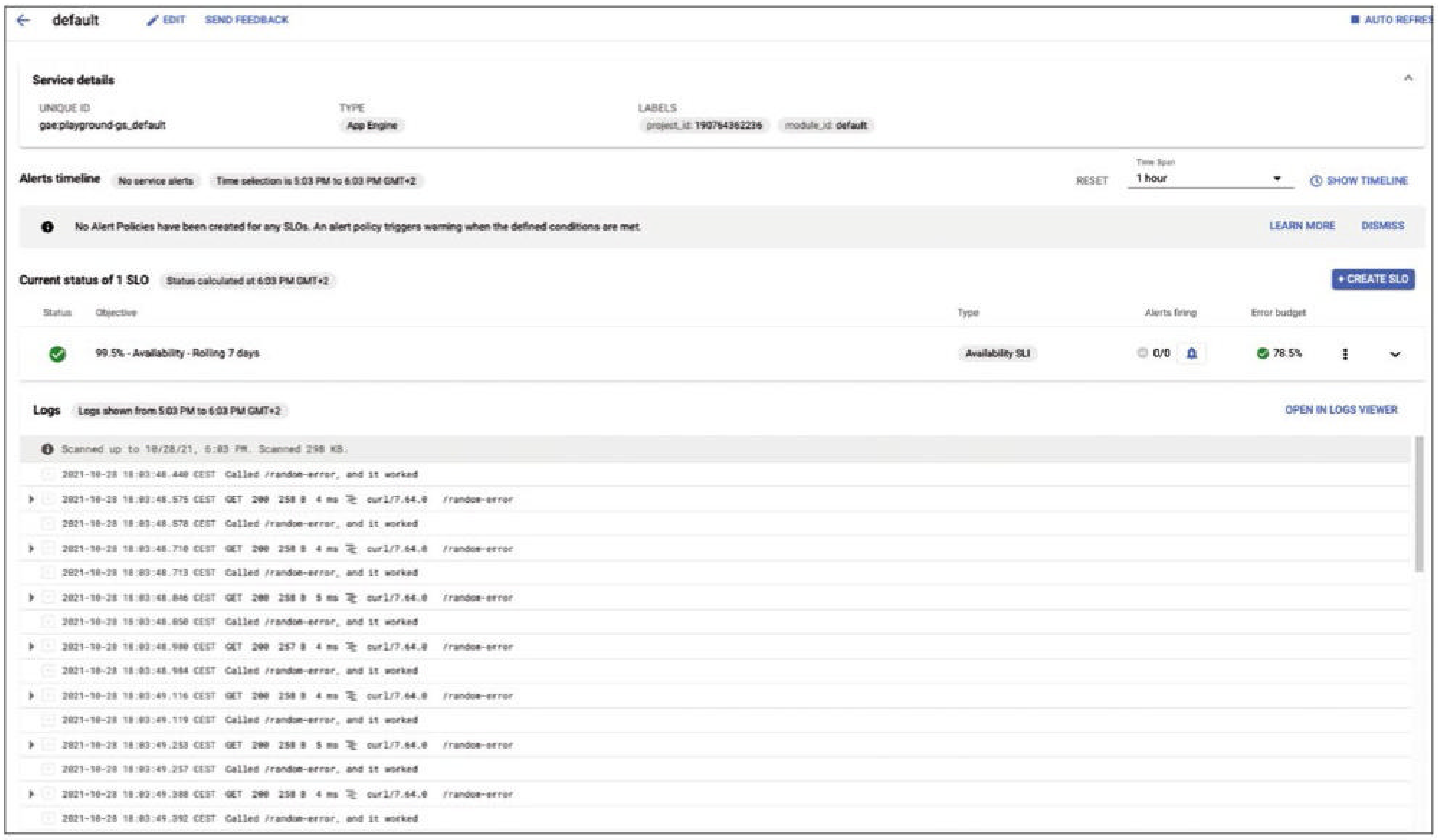
To select the logs, select *Subnetwork* in *Resource type* on the left, choose *compute.googleapis.com/vpc_flows* as the log name, and then filter by the desired subnet. If you want to evaluate the data with the help of SQL commands, you have to switch logging to legacy mode and then set up a log sink. Next, configure *BigQuery* – the serverless data warehouse from GCP – as the target. Of course, you can also work with log-based metrics by selecting the *Logs-based metrics* page in the menu on the left and then setting up a metric for the desired filter.

Conclusions

Google offers a comprehensive set of monitoring tools in its GCP platform that have now reached a high level of maturity. Therefore, efficient monitoring can be set up for the public cloud without the use of third-party tools. ■

Info

[1] Hello Logging NodeJS: [https://github.com/haggman/HelloLoggingNodeJS]



Service details

UNIQUE ID: gae:playground-gs_default

TYPE: App Engine

LABELS: project_id: 190764362236 module_id: default

Alerts timeline No service alerts Time selection is 5:03 PM to 6:03 PM GMT+2

No Alert Policies have been created for any SLOs. An alert policy triggers warning when the defined conditions are met.

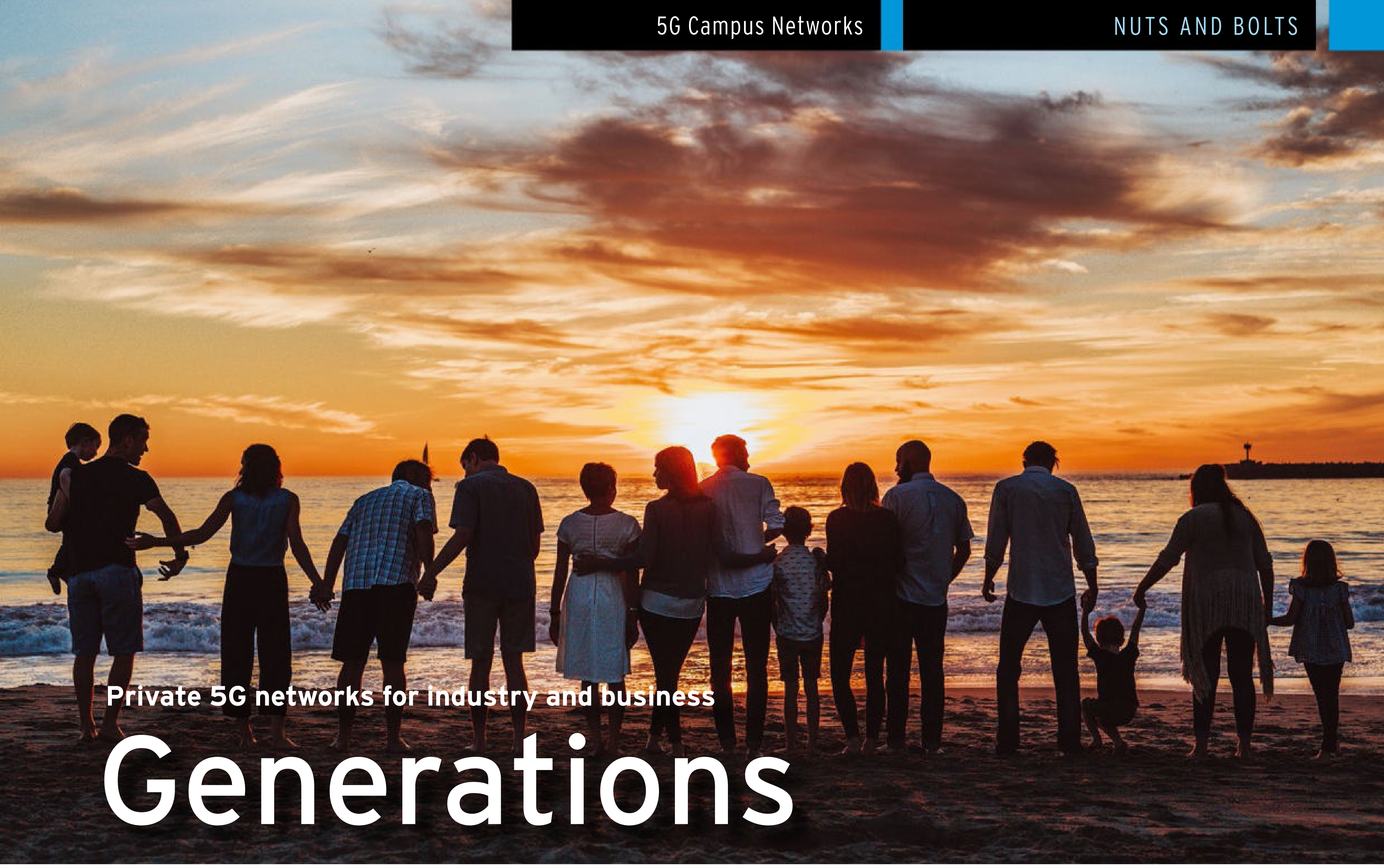
Current status of 1 SLO Status calculated at 6:03 PM GMT+2

Status	Objective	Type	Alerts firing	Error budget
✓	99.5% - Availability - Rolling 7 days	Availability SLI	0/0	✓ 78.5%

Logs Logs shown from 5:03 PM to 6:03 PM GMT+2

```
Scanned up to 10/28/21, 6:03 PM. Scanned 298 KB.
2021-10-28 18:03:48.440 CEST  Called /random-error, and it worked
2021-10-28 18:03:48.575 CEST  GET  200  258 B  4 ms  curl/7.64.0   /random-error
2021-10-28 18:03:48.578 CEST  Called /random-error, and it worked
2021-10-28 18:03:48.710 CEST  GET  200  258 B  4 ms  curl/7.64.0   /random-error
2021-10-28 18:03:48.713 CEST  Called /random-error, and it worked
2021-10-28 18:03:48.846 CEST  GET  200  258 B  5 ms  curl/7.64.0   /random-error
2021-10-28 18:03:48.850 CEST  Called /random-error, and it worked
2021-10-28 18:03:48.980 CEST  GET  200  257 B  4 ms  curl/7.64.0   /random-error
2021-10-28 18:03:48.984 CEST  Called /random-error, and it worked
2021-10-28 18:03:49.116 CEST  GET  200  258 B  4 ms  curl/7.64.0   /random-error
2021-10-28 18:03:49.119 CEST  Called /random-error, and it worked
2021-10-28 18:03:49.253 CEST  GET  200  258 B  5 ms  curl/7.64.0   /random-error
2021-10-28 18:03:49.257 CEST  Called /random-error, and it worked
2021-10-28 18:03:49.388 CEST  GET  200  258 B  4 ms  curl/7.64.0   /random-error
2021-10-28 18:03:49.392 CEST  Called /random-error, and it worked
```

Figure 3: The service dashboard after creating an SLO.



Private 5G networks for industry and business

Generations

The high speed and low latency of 5G is predicted to add value to industrial manufacturing, logistics, education, and data-heavy advanced applications. By Harald Kraft

Communications networks play an important role in digital transformation. With the rapidly increasing number of intelligent machines and industrial plants and of networked processes and systems, considerably more bandwidth is needed than before. In the corporate environment, a private fifth generation (5G) campus network on the company's own premises (Figure 1) is an interesting alternative to wireless local area or public networks (see the "Campus Networks" box). The high-performance transmission standard is a great fit for the demanding communications infrastructure of a networked factory.

However, 5G is more than just an innovative wireless technology with an expanded frequency spectrum. In combination with the Internet of Things (IoT) and artificial intelligence (AI), 5G looks to promote the development of new services and business models that would be virtually inconceivable with WiFi or wired networks, primarily because of its fundamental characteristics, simple frequency allocation, and moderate costs.

Wireless technology is expected to enable an almost unimaginable terminal device density of 1 million devices per square kilometer, which means it can connect an extremely large number of people, objects, and devices. Data is the fuel for digital

Campus Networks

Campus networks are geographically limited local mobile networks that can be deployed on both 4G and 5G technology. A 5G campus network is designed to meet special requirements (e.g., to allow Industry 4.0 [1] applications to communicate). The market for 5G campus networks is still in its early stages, but "the global private 5G network market size … is expected to expand at a compound annual growth rate of 47.5% from 2022 to 2030" [2].

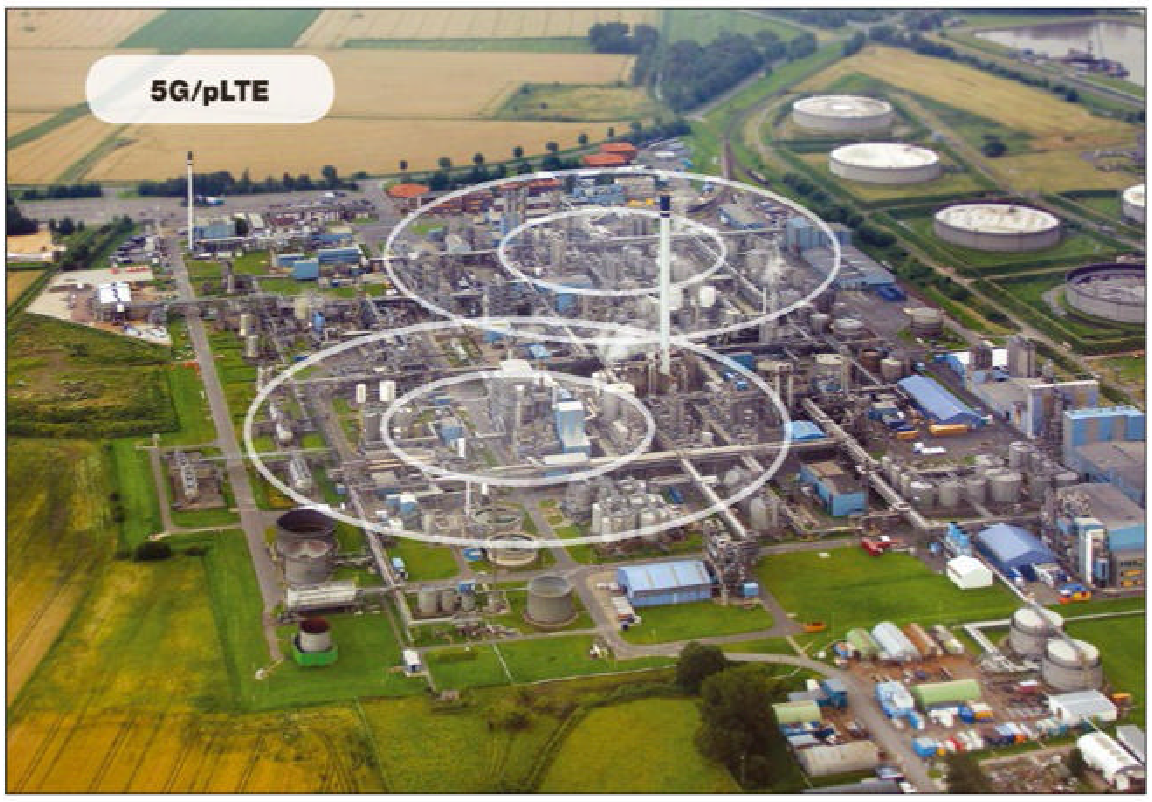


Figure 1: For the first time, 5G licenses enable private companies to set up their own campus networks.

Photo by Tyler Nix on Unsplash

transformation, and 5G will transmit data at up to 10Gbps in the future – which is 100 times faster than the current 4G/LTE standard. Additionally, the transmission delay (i.e., latency) is extremely low, which in turn keeps the response time for actions low.

In the future, 5G will be able to achieve latency times of less than 1ms. That is less than the blink of an eye and a crucial prerequisite for innovative developments such as autonomous driving. High availability is an elementary requirement for business-critical applications in particular, and because of 5G's system architecture, the system is particularly fail-safe, achieving operational reliability of 99.999 percent uptime.

Flexible Customization

For the first time, 5G licenses give private companies the ability to build proprietary campus networks. The motivations range from the desire to implement future-oriented applications that would not be feasible with conventional transmission technologies (or where a great deal more overhead would otherwise be involved) to the need to install a network tailored to one's own needs that can also be flexibly adapted to new requirements in the future.

A wide spectrum of application scenarios are possible, whether in production or on the construction site, in logistics, or in energy supplies.

An industrial production site, for example, would have IoT applications for transmitting sensor measurement data, augmented reality for plant maintenance and to provide guidance for repairs, and drones for monitoring the site or checking emission levels. Because 5G also minimizes wiring overhead and requires significantly fewer access points, production layouts can be redesigned to be far more flexible, which is likely to be necessary in the future, with markets and customer requirements changing at an increasing pace.

Another application of the 5G network is automated guided vehicles (AGV) to transport goods. This capability goes beyond classic fleet management for industrial trucks, such as those already implemented today with 4G or WLAN, to autonomous systems, whose numbers will increase rapidly in the future and whose tasks, such as autonomous route computation, will become far more complex.

Step-by-Step Expansion

The performance parameters of 5G are not yet fully available for all application scenarios because they are being developed step-by-step in a series of releases. Most of the performance features are currently based on Release 15, the enhanced Mobile Broadband (eMMB). This release aimed at extremely broadband communication by boosting data rates in the mobile communications system – with peak rates of up to 10Gbps and data volumes of 10Tbps per square kilometer. In June 2020, ultra-reliable and low-latency communications (URLLC) was added in Release 16 to improve latency to less than 1ms for real-time applications. Next up is expected to be massive machine-type communications (mMTC) in Release 17 in 2022 for an extremely high number of IoT end devices per unit area. The dates given are those for the specifications; however, network infrastructures and terminal devices are always subject to a delay because development and tuning to the respective release takes 15 to 18 months on average.

The advantage of 5G is the lightning-fast, delay-free, and secure exchange of data between many users and the associated use cases. To benefit, a holistic approach is essential when implementing a 5G campus network. The work starts with a comprehensive requirement analysis that takes the business objectives into account. In the beginning, it is essential to clarify which specific applications will help achieve these goals and how the information required for this purpose will reach the applications from machines, motors, production control, or video surveillance. Once all parameters are on the table, a reliable statement on whether a legacy WiFi or wired network, 5G with its new performance features, or a mixture of both is the best solution for a company.

Formula for Licensing Fees

Getting onboard with a private 5G network will differ according to where you live [3]. In the US, for example, you can have licensed or unlicensed spectrum managed by the enterprise or a service provider. The Citizens Broadband Radio Service (CBRS) spectrum band [4] is a 150MHz portion of radio spectrum from 3.5 to 3.7GHz for 5G wireless networks made available by the US Federal Communications Commission (FCC) for private use. "Private entities can access the CBRS band to build enterprise infrastructure services without expensive licensing fees" [5]. In tier 2, enterprises can acquire by auction priority access licenses (PALs) of 10MHz channels that can be renewed after 10 years, whereas tier 3 allows free access, although it requires registration with the Spectrum Access System (SAS). In this case, when the enterprise installs an access point, the device checks with the centralized SAS repository of all CBRS users and asks for the use of a specific channel. In this way, the repository helps avoid interference among devices.

In Germany, on the other hand, a property owner or tenant, with the owner's consent, can submit an application for frequency assignment. A joint application by several property owners for an area is also possible. Allocation by the German Federal Network Agency also takes place in 10MHz increments for a maximum period of 10 years. The fees in the 3.7 to 3.8GHz frequency range can be calculated by a simple formula, in which only the bandwidth, the period, and the area need to be entered as variables.

For example, if you have a site measuring half a square kilometer, the cost of 100MHz over 10 years is EUR16,000. The costs are very moderate, which makes 5G interesting for smaller companies. Additionally, companies in agriculture, forestry, or industry in less densely built-up areas can be subsidized. The EU and federal governments or states provide additional funding for a whole catalog of topics.

The architecture of a 5G campus network is independent of the system manufacturer and basically comprises the same building blocks. The wireless access network connects the end devices and enables them to access the independent core network. The end devices run various applications that are either installed locally on campus or located in a cloud.

The implementation of a campus network can basically be broken down into five phases: (1) planning and (2) network design, including (3) the frequency license application, followed by (4) the network installation and (5) commissioning phase. Setting up a campus network on your own requires a great deal of expertise in the fields of network technology and wireless communications.

Turnkey private campus networks are offered by systems integrators who can draw on appropriate experience in the planning, assembly, commissioning, and maintenance of communications networks. They integrate the 5G technology with the existing IT and production network. The crucial factor here is smooth integration with existing information and the communications technology system landscape so that production and processes can continue to run without disruption – after the 5G network has gone live, as well.

Regardless of whether a company implements a private campus network on its own or commissions a service provider, the system components require regular maintenance during ongoing operation. In addition to technical support and software updates, new issues also appear on the agenda (e.g., SIM card and terminal device management) because of the large number of end devices.

Network Security

As advanced as 5G is, the technology still faces challenges in the area of network security. Many attack vectors open up for criminals when IoT solutions based on 5G technology are implemented that involve a very large number of terminal devices communicating on the network. To prevent attacks by cybercriminals, companies need to establish a security strategy right from the start and reliably implement requirements that protect subscribers and use end-to-end encryption and network segmentation.

The zero trust strategy is ideal, in that it basically makes no distinction between users, devices, and services inside and outside your network. In principle, it trusts no one, checks all data traffic, and requires that all participants authenticate. In practice, companies often lack the personnel capacity or the know-how to implement security effectively. This gap can be closed by automating security, shifting the perspective away from "security by function" to holistic "security by design."

This principle additionally integrates security into processes with written definitions (e.g., for risk analysis or quality assurance) and maps roles, responsibilities, and activities within the organization, as well as the required technologies. Such a well thought out security concept is required from the outset, starting from the designated task of the technology, through analysis of the prerequisites for process-compliant applications by the user, among other things. If companies keep this necessity in mind, they can use 5G technology to build their own corporate campus network that is precisely tailored to future-proof IoT applications and their individual security needs.

Conclusions

5G campus networks offer added value not only in industrial manufacturing and logistics but also in many other areas. For example, schools, universities, and other educational institutions can expand digital learning with fast connections and high data volumes, as well as explore advanced application concepts of 5G in cooperation with partners from industry.

The same applies to agriculture and forestry, where more data on weather and the climate can be collected to deploy machinery in a more targeted way. There are virtually no limits to what you can imagine happening when it comes to application scenarios, because many users can benefit from private 5G networks thanks to the low latency, high data rates, and low susceptibility to interference. ■

Info

[1] Industry 4.0: [https://en.wikipedia.org/wiki/Fourth_Industrial_Revolution]

[2] 5G market analysis: [https://www.grandviewresearch.com/industry-analysis/private-5g-network-market]

[3] Global update on 5G, March 15, 2022: [https://www.qualcomm.com/videos/global-update-5g-spectrum]

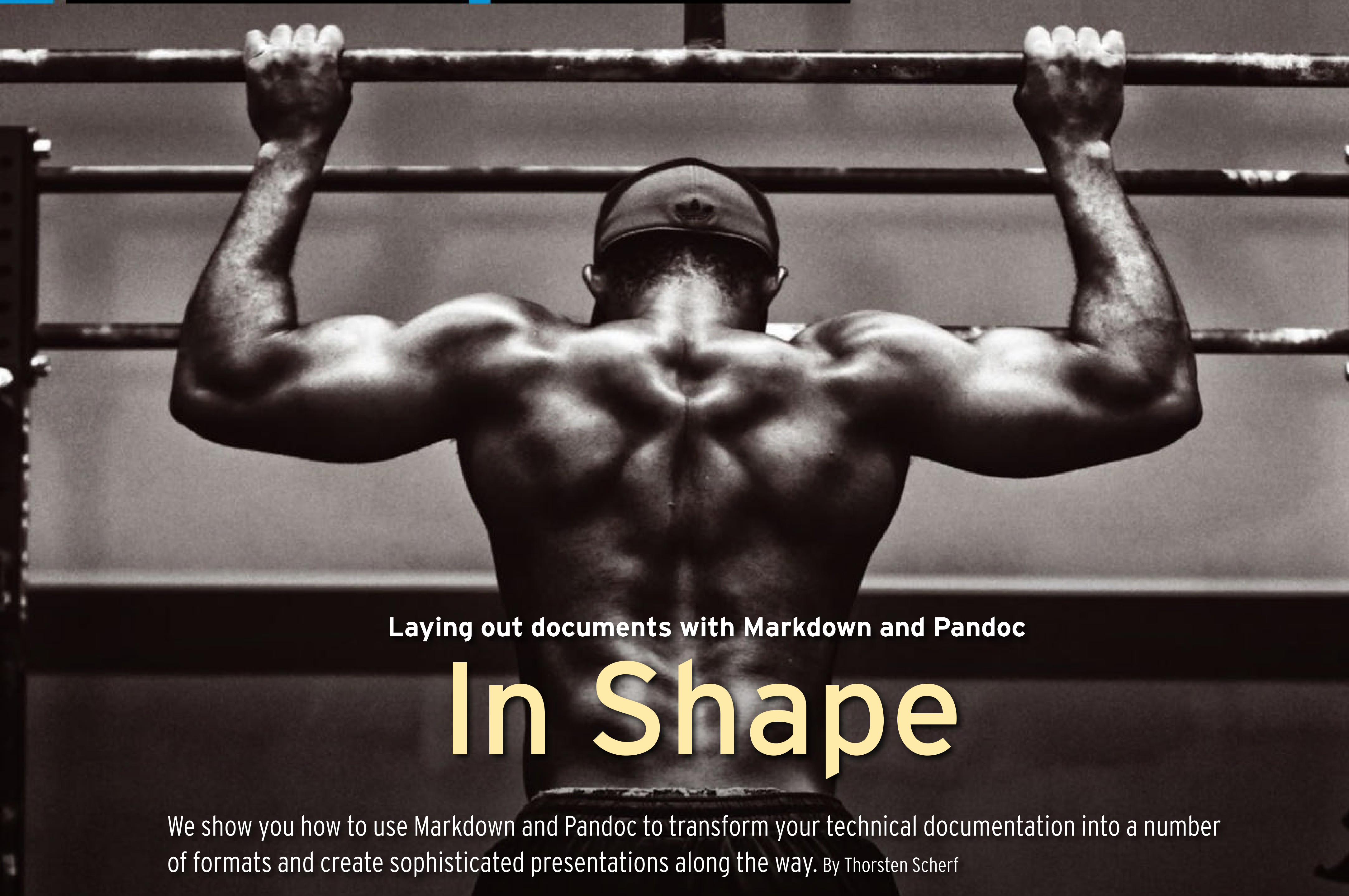
[4] Enterprise CBRS: [https://www.techtarget.com/searchnetworking/feature/Best-practices-for-enterprise-CBRS-deployment]

[5] Private spectrum and CBRS: [https://www.celona.io/cbrs/cbrs-5g]

Author

Harald Kraft is a network design consultant for private LTE/5G Networks/IoT at telent GmbH.

■



Laying out documents with Markdown and Pandoc

In Shape

We show you how to use Markdown and Pandoc to transform your technical documentation into a number of formats and create sophisticated presentations along the way.

By Thorsten Scherf

Text editing programs such as Microsoft Word, Apple Pages, and the open source LibreOffice suite can be used to create documentation and instructions. All of these programs mix different elements of a document together. For example, good documentation or a good manual does not consist of just plain text; it also contains images, graphics, tables, lists, and other elements. Conversely, this means that when writing, attention must be paid not only to the content of the document but also to coordinating the individual elements neatly in terms of the layout.

Unsurprisingly, this what-you-see-is-what-you-get (WYSIWYG) principle is very popular. Another type of tool maintains a division between the content of a document and the layout (e.g., in web development). No one would think of mixing the layout of a web page with the page content. Instead, a style sheet is used for the layout, typically available as a CSS (cascading style sheet) file [1]. To create a layout for the web page, you only need to modify the CSS, without having to change any of the page content.

TeX, LaTeX, and Markdown

If you prefer a logical division between content and layout, you can always use text typesetting systems such as TeX [2] or LaTeX [3]. However, these tools are complex, with syntax that is not necessarily intuitive, and require a certain period of familiarization. LaTeX is, strictly speaking, only a collection of macros that are intended to simplify the use of TeX.

Markdown [4] is far easier to use. This simple markup language can be learned, and therefore also used, by anyone in a short amount of time. Another advantage is that you can write a Markdown document in any text editor – you do not need a special program. For an easy introduction to the syntax of the language, I recommend you take a look at the Markdown Guide [5].

To generate a finished document from a Markdown text, you can use the Pandoc [6] tool. The software comprises a library written in Haskell and a command-line tool. These aids let you convert a variety of different text formats into other formats (e.g., Markdown into HTML, ePub, or PDF). The tool also supports formats such as DOCX and PPTX (Microsoft Word and PowerPoint).

These conversions are made possible by the use of a reader and writer, which are available for each format. A reader converts the text into an abstract syntax tree (AST) before a writer then transfers the elements from the AST into the appropriate target format. In some cases, the text has to go through an intermediate step. For example, to convert Markdown to PDF, a LaTeX version is first created, which is then converted to the PDF format with the `pdflatex` tool.

Installing Pandoc

The Pandoc software is available in the repositories of almost all Linux distributions. You will also find installation archives for Linux, Windows, and macOS on the Pandoc project site [6]. For Macs, you can install the software with the Brew package manager, as well. It is important to mention here that you need a TeX environment if you want to convert text to PDF. For macOS, this is easily done with Brew

Photo by Edgar Chaparro on Unsplash

by installing the *basictex* package. For Windows, the open source MiKTeX [7] distribution of TeX is available.

Web Pages in Markdown

In the following examples, I look at a couple of practical uses of Pandoc. The first task is to convert a Markdown text into (X)HTML so that you can publish it on a web page:

```
pandoc -M title="pandoc" ➘
  -s foo.md -o foo.html
```

The `-M` option lets you define arbitrary metadata for the output format – in this case, the title of the web page. The `-s` option ensures that only a single file is written as output, which means the stylesheet information is also contained in the header of the HTML file. A simple Markdown text is shown in Listing 1 that produces the output in Figure 1. The situation is very similar if you want to convert the Markdown text into a PDF:

```
pandoc -s foo.md -o foo.pdf
```

Pandoc tries to detect the input and output formats by referencing the file extensions. However, you can also specify the formats explicitly:

```
pandoc -f markdown -t pdf foo.md -o foo.pdf
```

(i.e., with `-f`/`--from` and `-t`/`--to`).

Creating Presentations

Another interesting option for Markdown with Pandoc is creating sophisticated presentations by taking one of several approaches. For example, you can create an HTML-based slide set on the basis of a JavaScript web framework or a PDF set that uses the LaTeX `beamer` class [8]. In terms of web frameworks, Pandoc supports S5 [9], DZSlides [10], Slidy [11], Slideous [12], and reveal.js [13].

To create a new presentation, call Pandoc in the usual way. For a PDF slide set, the command is:

```
pandoc -t beamer slides.md -o slides.pdf
```

If you would rather create an HTML slide set based on one of the previously mentioned JavaScript web frameworks instead, the command is only marginally different:

```
pandoc -t revealjs slides.md ➘
       -o slides-reveal.html
```

In the last example, make sure the JavaScript web framework is in the same directory as the Markdown file. Alternatively, you can specify a different path to the framework with the `-V revealjs-url` option.

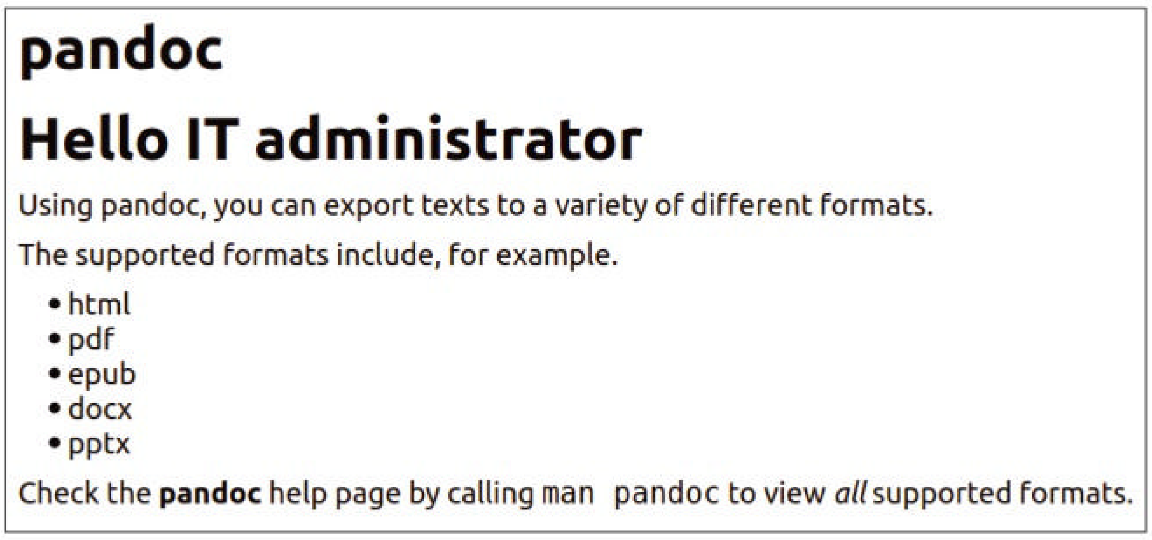
Of course, you can use a variety of other options, such as integrating different themes. The documentation for the respective frameworks contains all the necessary information.

Conclusions

To create appealing text documentation, you don't necessarily have to resort to classics such as Word or LibreOffice. A combination of Markdown and Pandoc lets you convert text into a number of different formats. PDF slide sets can also be created, as long as you have a LaTeX distribution installed on your system along with the `beamer` class. For HTML presentations, you then can use various JavaScript web frameworks to start a slideshow in any browser. ■

Listing 1: Simple Markdown

```
# Hello IT administrator
Using pandoc, you can export texts to a variety of different formats.
The supported formats include, for example.
- html
- pdf
- epub
- docx
- pptx
Check the **pandoc** help page by calling `man pandoc` to view *all*
supported formats.
```



pandoc

Hello IT administrator

Using pandoc, you can export texts to a variety of different formats.

The supported formats include, for example.

- html
- pdf
- epub
- docx
- pptx

Check the **pandoc** help page by calling `man pandoc` to view *all* supported formats.

Figure 1: The output of a simple Markdown document.

Info

[1] Cascading Style Sheets (CSS): [https://www.w3.org/Style/CSS/]

[2] TeX text typesetting system: [https://www.tug.org]

[3] LaTeX macros: [https://www.latex-project.org]

[4] Markdown markup language: [https://daringfireball.net/projects/markdown/]

[5] Markdown Guide: [https://www.markdownguide.org]

[6] Pandoc: [https://pandoc.org]

[7] MikTeX: [https://miktex.org/download]

[8] LaTex beamer class: [https://github.com/josephwright/beamer/]

[9] S5: [https://meyerweb.com/eric/tools/s5/]

[10] DZSlides: [http://paulrouget.com/dzslides/]

[11] Slidy: [https://www.w3.org/Talks/Tools/Slidy2/#(1)]

[12] Slideous: [https://goessner.net/articles/slideous/]

[13] reveal.js: [https://revealjs.com]

The Author

Thorsten Scherf is the global Product Lead for Identity Management and Platform Security in Red Hat's Product Experience group. He is a regular speaker at various international conferences and writes a lot about open source software.

ADMIN Network & Security

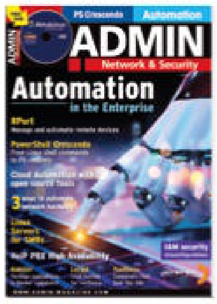
NEWSSTAND

Order online:
bit.ly/ADMIN-Newsstand

ADMIN is your source for technical solutions to real-world problems. Every issue is packed with practical articles on the topics you need, such as: security, cloud computing, DevOps, HPC, storage, and more! Explore our full catalog of back issues for specific topics or to complete your collection.

#68 – March/April 2022

Automation in the Enterprise



Automation in the enterprise extends to remote maintenance, cloud orchestration, and network hardware

On the DVD: AlmaLinux 8.5 (minimal)

#67 – January/February 2022

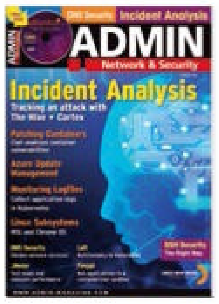
systemd Security

This issue, we look at how to secure systemd services and its associated components.

On the DVD: Fedora 35 Server (Install)

#66 – November/December 2021

Incident Analysis

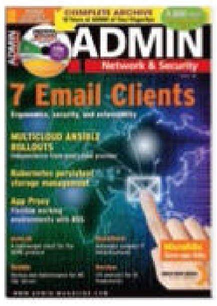


We look at updating, patching, and log monitoring container apps and explore The Hive + Cortex optimization.

On the DVD: Ubuntu 21.10 "Impish Indri" Server Edition

#65 – September/October 2021

7 Email Clients



The features in this issue tackle digital certificates, email clients, and HP backup strategies.

On the DVD: Complete ADMIN Archive DVD

#64 – July/August 2021

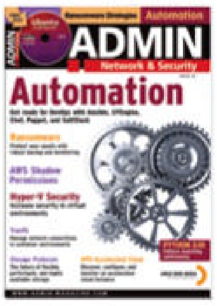
Bare Metal Deployment

Setting up, automating, and managing bare metal deployments gets easier with the tools presented in this issue.

On the DVD: Rocky Linux 8.4 (Minimal Install)

#63 – May/June 2021

Automation



This issue we are all about automation and configuration with some tools to lighten your load.

On the DVD: Ubuntu 21.04 Server

WRITE FOR US

Admin: Network and Security is looking for good, practical articles on system administration topics. We love to hear from IT professionals who have discovered innovative tools or techniques for solving real-world problems.

Tell us about your favorite:

- interoperability solutions
- practical tools for cloud environments
- security problems and how you solved them
- ingenious custom scripts
- unheralded open source utilities
- Windows networking techniques that aren't explained (or aren't explained well) in the standard documentation.

We need concrete, fully developed solutions: installation steps, configuration files, examples – we are looking for a complete discussion, not just a "hot tip" that leaves the details to the reader.

If you have an idea for an article, send a 1-2 paragraph proposal describing your topic to: *edit@admin-magazine.com*.

Authors

Thomas Bär	82
Joe Casad	8
Stefano Chittaro	62
Markus Feilner	28
Christopher Henkel	52
Ken Hess	3
Thomas Joos	72
Christian Knermann	66
Harald Kraft	91
Ankur Kumar	22
Martin Gerhard Loschwitz	10, 16, 42, 58, 76
Thorsten Scherf	94
Frank-Michael Schlede	82
Guido Söldner	86
Andreas Stolzenberger	36, 48
Jack Wallen	8

Contact Info

Editor in Chief
Joe Casad, jcasad@linuxnewmedia.com

Managing Editors
Rita L Sooby, rsooby@linuxnewmedia.com
Lori White, lwhite@linuxnewmedia.com

Senior Editor
Ken Hess

Localization & Translation
Ian Travis

News Editor
Jack Wallen

Copy Editors
Amy Pettle, Aubrey Vaughn

Layout
Dena Friesen, Lori White

Cover Design
Dena Friesen, Illustration based on graphics by rolffimages, 123RF.com

Advertising
Brian Osborn, bosborn@linuxnewmedia.com
phone +49 8093 7679420

Publisher
Brian Osborn

Marketing Communications
Gwen Clark, gclark@linuxnewmedia.com
Linux New Media USA, LLC
4840 Bob Billings Parkway, Ste 104
Lawrence, KS 66049 USA

Customer Service / Subscription
For USA and Canada:
Email: cs@linuxnewmedia.com
Phone: 1-866-247-2802
(Toll Free from the US and Canada)

For all other countries:
Email: subs@linuxnewmedia.com
www.admin-magazine.com

While every care has been taken in the content of the magazine, the publishers cannot be held responsible for the accuracy of the information contained within it or any consequences arising from the use of it. The use of the DVD provided with the magazine or any material provided on it is at your own risk.

Printed in Nuremberg, Germany by hofmann infocom GmbH.

Distributed by Seymour Distribution Ltd, United Kingdom

ADMIN is published bimonthly by Linux New Media USA, LLC, 4840 Bob Billings Parkway, Ste 104, Lawrence, KS 66049, USA (Print ISSN: 2045-0702, Online ISSN: 2831-9583). May/June 2022. Periodicals Postage paid at Lawrence, KS. Ride-Along Enclosed. POSTMASTER: Please send address changes to ADMIN, 4840 Bob Billings Parkway, Ste 104, Lawrence, KS 66049, USA.

Published in Europe by: Sparkhaus Media GmbH, Bialasstr. 1a, 85625 Glonn, Germany.

Systemmonitor
42
0,8
20
Profileinstellungen
Default custom profile
TUXEDO