Data Security and Storage Hardening In Rook and Ceph

Federico Lucifredi, Sage McTaggart – IBM

me! me! me!

Things I worked on

Ceph Storage
Ubuntu Server
Landscape
SUSE Studio
SLES
SMT
Ximian Red Carpet
Man (I)

me! me! me!

Things I worked on

Ceph Storage
OpenShift Data Foundation
Incident Response
Computer Security
Theoretical and practical

CEPH IN BRIEF

Ceph

- The future of storage!
- File, block and object storage
- Highly resilient
- Highly available
- Scale out

...absolutely awesome.

ROOK IN BRIEF

Rook

- Cloud-native storage for k8s
- Ceph-based: hyperscale
- Storage on top of compute: hyper-converge...

  ...or optionally external storage

- Automated resource management w/operators
- Automated upgrade and rollback

20 YEARS OF COMMUNITY-DRIVEN INNOVATION

"The Linux of Storage"

Enterprise-trusted, Community-developed

2007	2008-11	2012	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023
Sage Weil's UCSC dissertation Development funded by U.S. National Labs	Dreamhost Incubation Kernel merge	Ceph Argonaut First stable release Inktank launched	Ceph Firefly Erasure coding Red Hat acquisition	Ceph Hammer OpenStack dominance "Petabyte Release"	CephFS stable Bluestore Experimental Docker Container	Bluestore stable CephFS supported ISCSI, Ceph Metrics	Ceph Mimic RGW "Beast" stable Ceph Foundation	Rook 1.0 1+ billion objects ELS Lifecycle	Dashboard CephAdm Ceph ODF FIPS 140-2	RBD mirrored snapshots 10+ Billion objects NFS	Intel Optane Day 2 management Disk replacement UX	IBM Storage Ceph Internal QoS RGW multisite at scale

Vibrant open source developer community

- 1000+ contributors
- 200+ organizations
- 600K+ lines of code changed
- 17,000 code commits

Vibrant open source user community

- CERN
- NASA
- Bloomberg
- Flipkart
- Salesforce

4.2 to 6.5 EB deployed globally

THREAT MODEL

- Identify threat actors
  - Nation states
  - Organized crime
  - Hacker groups
  - Motivated individuals
  - Privileged insiders
  - Script kiddies
  - ...

NETWORK SECURITY ZONES

- Public Zone
  - **not** the public_network in Ceph
- Ceph Client Zone
- Storage Access Zone
  - public_network in Ceph
- Ceph Cluster zone

NETWORK SECURITY ZONES

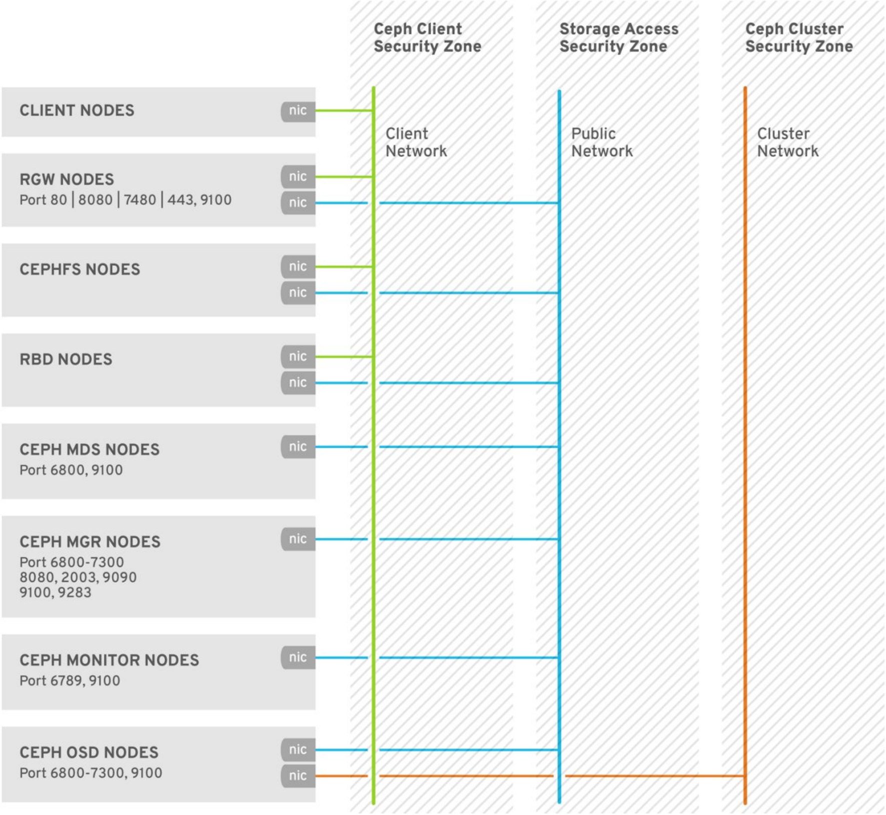
- Public Zone
    - **not** the public_network in Ceph
- Ceph Client Zone
- Storage Access Zone
    - public_network in Ceph
- Ceph Cluster zone

NETWORK SECURITY ZONES

- Public Zone
  - **not** the public_network in Ceph
- Ceph Client Zone
- Storage Access Zone
  - public_network in Ceph
- Ceph Cluster zone
  - cluster_network in Ceph

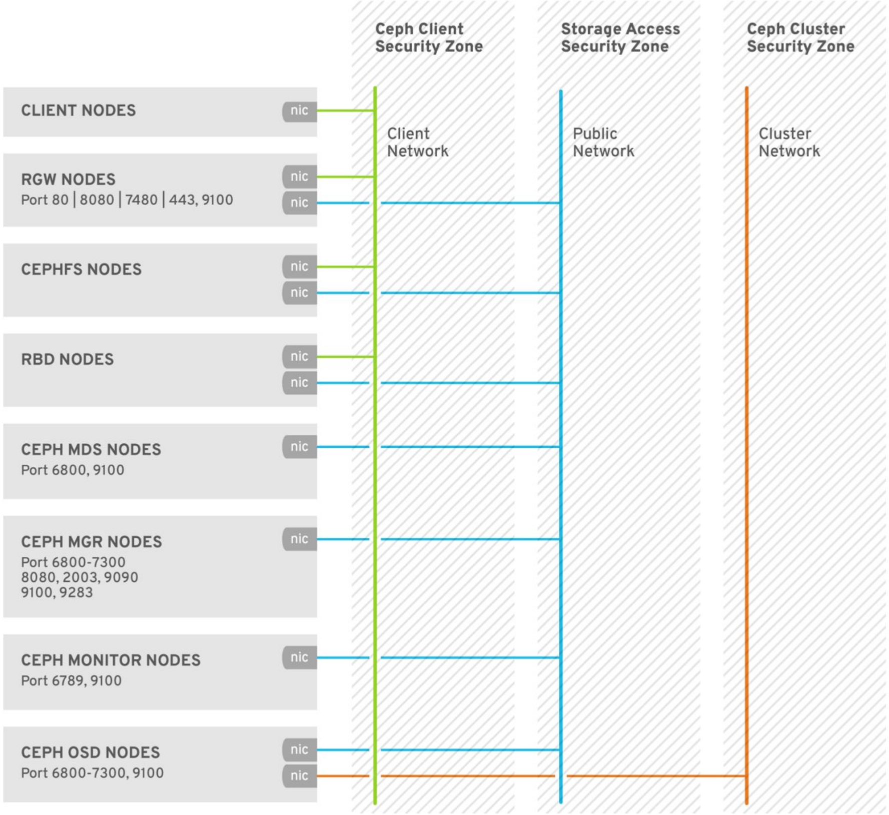
CONNECTING SECURITY ZONES

- Public Zone
  - **not** the public_network in Ceph
- Ceph Client Zone
- Storage Access Zone
  - public_network in Ceph
- Ceph Cluster zone
  - cluster_network in Ceph



CONNECTING SECURITY ZONES

- Public Zone
  - **not** the public_network in Ceph
- Ceph Client Zone
- Storage Access Zone
  - public_network in Ceph
- Ceph Cluster zone
  - cluster_network in Ceph



Ceph Client Security Zone

Storage Access Security Zone

Ceph Cluster Security Zone

Client Network

Public Network

Cluster Network

CLIENT NODES

RGW NODES
Port 80 | 8080 | 7480 | 443, 9100

CEPHFS NODES

RBD NODES

CEPH MDS NODES
Port 6800, 9100

CEPH MGR NODES
Port 6800-7300
8080, 2003, 9090
9100, 9283

CEPH MONITOR NODES
Port 6789, 9100

CEPH OSD NODES
Port 6800-7300, 9100

MOVING COMPANIES AS AN OSS PRODUCT

- The security of a product is good, but what about the support of a product?
  - An abandoned product can be forked, but do you really want to maintain it and patch it?
- We moved to IBM approximately a year ago from Red Hat
  - Definitely a change, but overall going well
- I will discuss some aspects of IBM Product Security, and how that's been going
- I will also discuss some of our accomplishments, and what we are planning going forward

PRODUCT SECURITY

- Product Security at IBM performs SDL activities across the life-cycle of each release
  - Goal to reduce risk and improve the security of Ceph
- Doing more formal pen tests and threat models
  - Improving every year
- Many new things coming with collaboration!
  - Goal is to work with IBM and spread open source in industry
  - This includes finding vulnerabilities in upstream Ceph and fixing them

IBM PSIRT

- We have manifested and documented all dependencies within both IBM and RH’s PSIRTs
- Automated many security scans of our code
  - Many customers want clean scans, so we’re working there to reduce even low risk vulns, in addition to compliance
- We’re onboarded with IBM PSIRT, and are fixing all prior vulnerabilities, no matter how low risk
- We are working on automatically updating all/most dependencies
  - Isn’t trivial, often breaks builds
  - Prioritizing libraries with many CVEs per release-reduces exposure
  - Next new releases will auto-update Grafana to latest minor

IBM PSIRT

- We will continue to review existing vulns, and release regular security updates, and improve code security preemptively
  - Have gotten more upstream engagement, hundreds of vulns, within past year
  - Have implemented changes, e.g, to container updates, based on requests
- We will follow IBM standards to fix vulns and ensure compliance this year
  - Currently we’re fixing all CVEs, and aim to be fully within IBM SLA/regs by EOY
  - Working on automatically updating problem libraries
- Have pipeline to automate code scans to detect new vulns
  - Also scans dependencies and records them

NEW WORK AT IBM

- All of the CVE fixes will be continue to be ported to upstream Ceph. We aim to keep all versions of Ceph, be that Red Hat, IBM or Upstream, equally secure
- New collaboration produces new challenges, and lots of goals. Going well!
- Working to improve dashboard security
- Worked to improve Call Home functionality and security for IBM support by applying Open Source principles
    - If it stinks, other people will smell it. Visible bugs are a boone
- Collaboration with IBM open source and licensing should spread OS!
- Working on use as backend for AI, govt clients, etc.
- Reach out to talk about what you want to see with our collaboration with IBM, and feel free to discuss any concerns with us

New Cryptography Goals

- Working with IBM research, discussing quantum and confidential computing
  - Currently determining viability, but has been productive to see where we use cryptography
  - Open source research code implemented into an open source storage system is the goal!
- Documenting all places we use cryptography and if it is quantum safe(er)
- Via architectural diagrams, working to see best places for TCMs
- Priority is to have all work on this be open source and use modern open source libraries here

ENCRYPTION AND KEY MANAGEMENT

- Data at rest (OSD)
  - OSDs can be encrypted with dmcrypt at creation.
  - Write-ahead logs, journals and metadata stores can also be secured
  - LUKS provides a variety of cryptographic options
  - All data at rest is encrypted irrespective of access protocol
  - FIPS 140-2 certified cyphers can be used
- Encryption keys
  - Stored in the Monitor daemon (MON) – or KMS, or K8s Secrets
- Object Gateway (RGW)
  - Data is encrypted at rest relying on OSD strategy
  - Alternatively, data can be encrypted at ingestion with locally managed keys
  - Keys can be managed externally with HashiCorp Vault KMS
  - OpenStack Barbican and KMIP-compatible KMS support is also available

ENCRYPTION IN TRANSIT

- Data in transit
    - Ceph’s internal protocol can be encrypted as a Messenger v.2.1 protocol option
    - Legacy cleartext protocol is still default for compatibility reasons
    - All data at rest is encrypted irrespective of access protocol
    - FIPS 140-2 certified cyphers can be used
- Client and public security zones
    - TLS security can be used from Object Gateway to S3 clients.
    - TLS termination at HAproxy a special case
- Network hygiene
    - Firewalld at individual nodes

ROOK SPECIFIC

- CRDs can be used to encode security preferences
  - Example: client configuration
  - Example: RGW certificate
  - Allows principle of least privilege to easily be implemented
- Rook provides at-rest data encryption as discussed
  - Setup of Msgr v.2 in-flight encryption exists as of 1.9
  - Use software-defined cloud network fabric to segregate traffic
- Standard k8s user permissions apply to persistent volumes
  - Nothing Rook needs to do here
- CSI driver supports KMS
  - PVs can be encrypted with individual keys, limiting key scope

CONTROL PLANE

- SSH
  - Cephadm, ceph-ansible and other tools
  - User (cephadm or ceph) with password-less root access can be used
  - Access is secured with SSH keys
  - Port 22
- Management Dashboard
  - TLS on port 443 (operator facing (storage access zone)
  - Dashboard access zone often tailored by operators to suit local threat model
  - Option for SAML authorization, Kubernetes native authorization
- Manager (MGR)
  - Ceph protocol on port range 6800-7300 (storage access zone)

IDENTITY AND ACCESS

- Cephx
  - Shared secret keys are in use for authentication
  - Mechanism protects cluster from MITM attacks
  - Authentication and authorization are on by default
    - If user is not supplied, provide client.admin as user, and restricted accordingly
- Object Gateway (RGW)
  - S3 user: access key and secret model, option for bucket policy
  - Swift user: access key and secret model
    - Note that default Swift user is sub-user of S3 user, deleting S3 user will delete the Swift user as well
  - Administrative user: access key and secret with access to administrative API
  - User authentication is stored in Ceph pools
    - Identity is an IAM API, consistent and secure
    - Token based authentication with STS API
  - Can couple with OIDC providers (Keycloak, etc), backed by organizational IdP (FreeIPA) for granular role or attribute access

IDENTITY AND ACCESS

- LDAP and Active Directory users can be used as identity services
  - Secure LDAP is highly recommended

- OpenStack Keystone
  - Ceph supports using OpenStack Keystone to authenticate Object Gateway users

AUDITING

Operator actions

- Stored in /var/log/ceph/ceph.audit.log

For example:

```
2018-08-13 21:50:28.727176 mon.reesi001 mon.0 172.21.2.201:6789/0
2097902 : audit [INF] from='client.348389421 -' entity='client.admin'
cmd=[{"prefix": "osd set", "key": "nodown"}]: dispatch

2018-08-13 21:50:28.872992 mon.reesi001 mon.0 172.21.2.201:6789/0
2097904 : audit [INF] from='client.348389421 -' entity='client.admin'
cmd='[{"prefix": "osd set", "key": "nodown"}]': finished
```

In distributed systems, actions may start on one node (dispatch) and propagate to others (finished)

DATA RETENTION

RADOS

- End users generally do not have the ability to read, write or delete objects directly in a storage pool

Ceph Block Device (RBD), Object Gateway (RGW), Filesystem (MDS)

- Users can create, delete, modify volume images, objects or files
- Deletion destroys corresponding RADOS object in unrecoverable manner
  - RBD pools may provide “trash bin” functionality with spare capacity
  - RGW bucket lifecycle supports versioning. Residual data artefacts may persist in storage medium

Secure deletion

- Sanitize retired media by encrypting the OSD contents at rest, and destroying the encryption key

INFRASTRUCTURE HARDENING

- SELinux
  - Red Hat Ceph storage clusters default to SELinux in enforcing mode
- FIPS 140-2 support
  - Certified cryptography can be imported in RHEL “FIPS mode” setup
  - RHEL 8.4 is the most recent certified version
- Hardened binaries
  - `-D_FORTIFY_SOURCE=2`
  - `-D_GLIBCXX_ASSERTIONS`
  - `-fstack-protector-strong`
  - `-fcf-protection`
  - `-fstack-clash-protection`
- Additional Kernel or OS-supplied hardening
  - `SECCOMP`
  - `PIE`
  - `RELRO`
  - `BIND_NOW`
  - `ASLR (all varieties)`

Thank you!

RESOURCES

- Managing and Securing Kubernetes Secrets
  - Rani Osnat - Aquia Security
- Hacking Kubernetes – chapter 6: Storage
  - Andrew Martin and Michael Hausenblas (O’Reilly)
- Data Security and Hardening Guide
  - IBM Storage Ceph and Red Hat Ceph Storage documentation
- Encrypting Secret Data at Rest
  - Kubernetes documentation
- Recommended Compiler and Linker flags for GCC
  - Survey of Kernel and Userspace hardening options

CREDITS

Federico Lucifredi
Sage McTaggart
Michael Hackett
John Wilkins
J.C. Lopez
Travis Nielsen
Sébastien Han
Ken Dreyer
Kyle Bader