SDC

STORAGE DEVELOPER CONFERENCE

SNIA SANTA CLARA, 2017

Mitigating Ransomware Attacks at the Block Level with OpenZFS

Michael Dexter

Gainframe, SNIA DPCO

What is Ransomware?

- Working SNIA Definition: A type of malicious software (malware) that prevents or limits users from accessing their system, applications, or data, or alternatively, to publish the user's data unless a "ransom" fee is paid
- CryptoLocker, CryptoWall
- WannaCry, Petya

What is Ransomware?

- Encryption of data
- Publication of data
- Prediction: Exfiltration of data or “Datanapping”
- Payment: Bitcoin, Premium SMS…
- “Phishing” bait, “You won’t believe…”
- Advertising networks

Ransomware Reach

- Popular file types
- Network shares
- Online backups
- Document previous versions/“Shadow Copies”
- Cloud accounts, “DropBox”

Universal Vector: Write Access

- Nefarious in their simplicity
- Indistinguishable from data deletion by users
- Behavioral detection cat and mouse
- Exfiltrate and delete are simply move & write 1X
- ***Self-inflicted, virtually no "Hacking" involved using user default write permissions***

“Just update your antivirus software”

- Write access is the primary attack vector
- Consider ‘sudo’ discreet privilege escalation
- Consider using web applications
- My informal poll reported *server-side attacks*
- *Restricting write permissions is the only file-level mitigation strategy*

Possible Warning Signs

- Out-of-space error as encrypted data replaces unencrpyted data
- High write activity from encryption activity
- Actual encryption activity via tracing
- Unusual data exfiltration

Out of Band: Mitigation at the Block Level

- System administrator territory by definition
  - “Superuser” privileges at the file level
  - “Superuser” device control at the block level
- The oldest, simplest computer security model
- Reasonably file system-agnostic
- In-place/internal to the file system

Block-Level Versioning via Snapshotting

- Unrestricted user actions mandate “undo” ability
  - Outside user default permissions/reach
  - Ideally non-destructive undo
  - Ideally fine-grained/per-user and local
- Requires clear, coordinated RPO/RTO

Block-Level Versioning via Snapshotting

- RPO: Recovery Point Objective
  - Undo “Levels”/Timeframe
- RTO: Recovery Time Objective
  - Admin! Teacher! Help! I deleted all my data!
  - Clear SLA and procedures with users
  - Does your support infrastructure scale?

Block-Level Versioning via Snapshotting

- Benefits beyond Ransomware mitigation
- Ransomware is the motivator of the hour
- Assumption of snapshotting abilities in your FS

Snapshotting File Systems

- FreeBSD UFS2 Snapshots
- GNU/Linux LVM Snapshots
- Dragonfly BSD HammerFS
- GNU/Linux Btrfs Snapshots
- NTFS Volume Snapshot Service/Shadow Copies
- WAFL and Oracle ZFS Snapshots

Snapshotting File Systems

- Generally bolted-on functionality
- Often with performance impacts
- Some fine-grained, some not
- Few desktop/server/NAS-agnostic options

Institutionalized Snapshotting: OpenZFS

- Copy-On-Write (COW)
  - Write and dereference, rather than overwrite
  - Organized by sequential Transaction Groups
  - Universal opportunity to snapshot
  - New data = deltas aside existing data

Institutionalized Snapshotting: OpenZFS

- Institutionalized snapshotting allows…
  - Fine-grained at dataset “File System” level
  - Writable snapshots in the form of Clones
  - Clones allow for forensic preservation
  - Promotable to independent File Systems
  - Foundation of OpenZFS replication

Other OpenZFS Features

- Institutionalized checksumming
  - Merkel tree, no “UPS problem”
- ZVOL synthetic block device support
  - Flexible size, quotas, block size
  - Foreign File System support
  - Local attached and iSCSI/FC availability

Other OpenZFS Features

- Open Source
- Cross platform/endian-agnostic
- Nestable Datasets for fine-grained control
- Supports “hybrid” flash read/write acceleration
- Highly flexible, unlimited snapshots
- Enough features for two books: zfsbook.com

OpenZFS in Practice: Operating Systems

- OpenSolaris come Illumos and derivatives
- FreeBSD and derivatives
- GNU/Linux with legal uncertainty
- macOS for data partitions

OpenZFS in Practice: Availability to Users

- Local File System and block device access
- Network File Sharing
  - SMB, NFS, AFP, FTP etc.
- Network block sharing or image on File System
  - iSCSI, FibreChannel, RAW IMG, VMDK etc.
- Unlimited client/guest Operating Systems

File and Block: Herein Lies the Flexibility

- Network File Sharing
  - Flexible Ransomware “undo” ability
  - Per-directory, per-user
- Network block sharing or image on File System
  - Per-LUN, per-virtual machine
  - Also mitigate unclean VM shutdown

Think About Your RPO and Retention

- When to Snapshot?
  - Daily? Hourly? Every five minutes?
  - Running out of space is resolvable
  - Losing historic granularity is *not*
  - During business hours?
  - Usage-driven shapshotting

Think About Your RPO and Retention

- Your RPO drives your snapshot frequency
- What retention policy?
  - The Long Holiday problem
  - “Backup” goals
  - Archiving obligations
  - Primary, secondary, tertiary storage?

Policy-Driven Technology

- Technical flexibility enables policy flexibility
- Talk to your users about their work habits
- Talk to your lawyers about retention obligations

Ransomware is a Wake Up Call
For Many Perennial Issues

Mitigating Ransomware In Practice

- Statistically requires an OS migration
- Many NAS/SAN appliance options
  - FreeBSD-Based: FreeNAS, QNAP
  - Illumos-Based: Syneto, Nexenta
  - GNU/Linux-Based: Datto

Mitigating Ransomware In Practice

- My Experience is with FreeBSD and FreeNAS
- Open Source solutions enable full support
- Broadest user feedback scope
- Culture of vendor and individual contribution
- Excellent overlap with SNIA activities

Regardless of the Platform You Choose…

- Establish and maintain redundancy
- Flexible and scalable RaidZ/stripe of mirrors
- Create Datasets based on policy/org chart
- Create ZVOL block devices as needed
- Determine a snapshot and retention policy
- Share your datasets and block devices

Regardless of the Platform You Choose…

- Periodic “scrubs” validate all data checksums
- Replaced failed storage devices as needed
- Watch their S.M.A.R.T. data
- Determine expected performance
- Recognize degraded performance

Red Alert!

- Communication comes first
  - Shortens Recovery Time
  - Stops the spread of the Ransomware
  - Helps prevents future infection
- Educate users avoid Ransomware
- Educate users recognize an attack

Red Alert!

- Infected systems will re-infect – cleanse them
- Clearly communicate what data is impacted
- Decide if forensic information is desirable
- Determine if critical data exceeded the Restore Point – adjust accordingly
- *Learn from every experience*

Under the Hood

```
zfs list -t snapshot

NAME                       USED  AVAIL  REFER  MOUNTPOINT
myvol/users@2017-09-10        0      -   780K           -

zfs snapshot myvol/users@2017-09-11
zfs clone myvol/users@2017-09-10 myvol/users@recover
zfs rollback myvol/users@2017-09-10
```


SDC

STORAGE DEVELOPER CONFERENCE

SNIA SANTA CLARA, 2017

Thank you!

@MichaelDexter
dexter@gainframe.com