halcyon | Ransomware Research Center

Pay2Key Linux: Purpose-Built Proxmox VM and Backup Destruction

A new strain of Iranian-linked ransomware is going after organizations that run their servers on Proxmox, an open-source platform that has become a popular alternative to VMware in recent years. Halcyon has identified a Linux version of the Pay2Key ransomware family that was built from the ground up to attack Proxmox environments. This version shuts down the virtual machines that run a company's applications, delete the backups that would normally allow recovery, and then encrypt what remains.

Proxmox is widely used by small and medium sized businesses and MSPs across South America, Europe (and heavily in eastern Europe), Southeast Asia, and Middle East. It is also used by some organizations during post-ransomware/remediation.

What makes this strain notable is that the Pay2Key attackers studied how Proxmox works and engineered their tool to use Proxmox's own management commands against it, including a deliberate bypass of the safeguards Proxmox uses to protect backups from deletion. No other ransomware family Halcyon tracks has modified built-in backup protections like this at the platform level.

Background

Pay2Key is a ransomware family first observed in late 2020, initially conducting rapid, coordinated attacks against organizations in Israel, primarily targeting Windows network infrastructures via compromised RDP endpoints. The FBI, CISA, and the Department of Defense Cyber Crime Center jointly assessed Pay2Key as an Iranian information operation in August 2024.

The group adapted since its emergence, expanding capabilities to target Linux and virtualized infrastructure to maximize disruption across heterogeneous enterprise environments. Renewed activity was documented in mid-2025, with researchers tracking at least 51 confirmed ransom payouts across a four-month stretch totaling more than $4 million. Since the summer 2025 campaign, the group has claimed more than $8 million in ransom payments tied to approximately 170 global victims. Pay2Key resumed its ransomware-as-a-service (RaaS) operation and raised its affiliate profit share from 70% to 80% in an effort to be competitive with other ransomware groups who offer similar to 80% like Qilin. Activity has historically correlated with geopolitical flashpoints involving Iran, Israel, and the United States.

The Linux variant analyzed in this report, a 64-bit ELF binary compiled in C++and first detected in late August 2025, represents a significant architectural upgrade from earlier iterations. It introduces hardened anti-forensics, watchdog-based persistence, and, most critically, purpose-built logic for Proxmox Virtual Environment that has no equivalent in other tracked ransomware families.

Why Proxmox and Why Now

Proxmox VE adoption accelerated following Broadcom's 2023 acquisition of VMware. Due to the surge of Proxmox migrations through 2024 and 2025, the platform now underpins significant production workloads globally. This migration wave created an opening that Pay2Key's operators identified and exploited.

The historical ransomware playbook for hypervisor attacks was built around VMware ESXi. ESXiArgs, BlackBasta and LockBit, among other ransomware actors, employed dedicated ESXi encryptors with `esxcli` based VM shutdown sequences. None of the ESXi-focused attack tooling translates to Proxmox. Proxmox's architecture differs fundamentally as it uses QEMU/KVM, manages VMs and containers (LXC) through a REST API exposed via the pvesh command-line tool, and stores cluster state in `/etc/pve`.

Pay2Key is the first widely observed variant to have engineered attack logic specific to this architecture, meaning organizations that migrated from VMware to Proxmox now face a threat actor that has explicitly targeted their platform.

Halcyon Ransomware Chain

Privilege Escalation

Pay2Key Linux requires root-level privileges to execute. If run without root access, the process exits immediately. Upon execution, the ransomware decrypts an embedded JSON configuration appended to the end of the binary using ChaCha20, with the decryption key derived from the binary's own footer bytes via bitwise NOT operations. The configuration governs all runtime behavior: encryption thresholds, file and directory filters, targeted processes and services, and master public key material. A hardcoded magic value (`ff fa 61 22 33`) validates the configuration header and if the magic bytes are absent, the malware exits.

Environment Enumeration

Proxmox Detection and Cluster Enumeration

Prior to commencing system-wide file traversal, Pay2Key performs hypervisor detection by checking for the presence of the `/etc/pve` directory, a path unique to Proxmox installations, which serves as the cluster configuration repository.

If Proxmox is confirmed, the ransomware calls the Proxmox REST API through the `pvesh` command-line tool to enumerate all cluster resources:

```
pvesh get /cluster/resources --output-format json
```

The JSON output is parsed to extract all `Proxmox::VM` (virtual machines), `Proxmox::Node` (cluster nodes), and `Proxmox::Storage` (storage repositories) objects. This gives Pay2Key a complete map of the entire virtualized environment without requiring any prior knowledge of the infrastructure layout. This Proxmox-specific enumeration logic is unique among all ransomware families tracked by Halcyon.

Security Bypass

Prior to initiating any encryption, Pay2Key executes a series of commands to disable prominent Linux security mechanisms, preventing security policies from blocking or logging the subsequent encryption routine:

Disable SELinux enforcement

- `setenforce 0`

Fully disable AppArmor

- `aa-teardown`
- `systemctl disable apparmor`
- `systemctl stop apparmor`

These commands collectively strip the host of its active security frameworks before the encryption routine starts.

Persistence

Upon execution, Pay2Key achieves persistence by installing a crontab entry that triggers the ransomware binary on every system reboot: `@reboot <path_of_ransomware>`. This is accomplished by writing to a temporary file and loading it via `crontab`. The entry ensures that even a detection-triggered reboot resumes encryption. Upon successful completion of encryption, this crontab entry is removed, eliminating the persistence mechanism after the attack concludes. Pay2Key also incorporates an advanced watchdog that uses PID files at `/etc/<hash_of_main>` and monitors process liveness via the `kill(pid, 0)` syscall, restarting the primary process via `fork()` and `execve()` if it is terminated.

Data Destruction

Forced VM and Container Shutdown

For each VM identified in the cluster inventory, Pay2Key issues a forced shutdown using the QEMU management tool with the `--skiplock` flag:

```
qm stop <vmid> --skiplock
```

For each Linux container (LXC), the equivalent command is used:

```
pct stop <ctid> --skiplock
```

The `--skiplock` flag is the operative detail. Proxmox implements lock files on VMs and containers during operations such as migrations, backups, and snapshots to prevent conflicting changes. The `--skiplock` flag overrides this protection, allowing Pay2Key to halt workloads that would otherwise be protected from modification with a deliberate, platform-aware bypass of a Proxmox-specific safety mechanism with no equivalent in any ESXi-focused ransomware.

Backup Destruction

After shutting down virtual workloads, Pay2Key targets Proxmox backups through a two-step API sequence. First, the protection flag on each backup is cleared:

```
pvesh set /nodes/<node>/storage/<storage>/content/<backup> --protected 0
```

Then the backup is permanently deleted via the cluster API:

```
pvesh delete /nodes/<node>/storage/<storage>/content/<backup>
```

This is the most operationally significant differentiator from all other tracked ransomware families. Proxmox allows backups to be flagged as “protected,” a safeguard specifically designed to prevent unauthorized deletion. Pay2Key bypasses this protection by first clearing the flag and then invoking the API-level delete. File-extension-based deletion by other ransomware families would not reach backups stored in managed Proxmox datastores, making this API-level approach uniquely effective against Proxmox environments. Organizations relying on Proxmox's built-in backup protection as a ransomware mitigation are specifically vulnerable.

Data Encryption

Process and Service Termination

Pay2Key terminates a comprehensive list of services and processes defined in its JSON configuration prior to encryption. Targeted categories include databases (MySQL, PostgreSQL, MongoDB, Oracle, Redis, Elasticsearch, and more than 25 others), backup and archiving tools (Veeam, Bacula, Bareos, rsync, Restic, Duplicati, and others), and virtualization services (libvirtd, QEMU, VMware tools, and XCP-ng/Xen components). Services are disabled and stopped via systemctl; processes are terminated via the `kill()` syscall. This ensures that open file handles are released, and data is flushed before encryption, maximizing the number of files that can be successfully locked.

File Enumeration

Before initiating file enumeration, Pay2Key executes `mount -a` to ensure all configured mount points are accessible. The ransomware then traverses the filesystem from root (/), applying a comprehensive exclusion list (system binaries, boot, /etc, /proc, /run, /dev, /sys, and others) to avoid rendering the host unbootable. ELF and MZ executables are explicitly excluded to avoid crashing the host mid-operation. Files matching prioritized extensions are encrypted first:

- VM Disk images: `qcow2, vma, vmdk`
- Databases: `sql, mdf, mdb`
- Backups: `bak, img, backup`

Pay2Key uses ChaCha20 for file content encryption with per-file keys protected via Curve25519 (X25519) asymmetric encryption, making recovery without the attacker's private key cryptographically infeasible. Two encryption modes are applied based on file size: full encryption for smaller files and intermittent block encryption for larger files, where the first and last portions are always fully encrypted, and the middle is encrypted at deterministic intervals. This randomization complicates both recovery and forensic analysis, as each infection produces unique encrypted segments. Encrypted files are renamed by appending a unique extension tied to the specific deployment.

C2 Reporting and Anti-Forensics

Pay2Key transmits JSON-formatted encryption status reports; start and finish events including hostname, platform, file count, and duration to numerous configured C2 servers over raw ICMP (using `SOCK_RAW, IPPROTO_ICMP`). The reports are XOR-encrypted with a key derived from a Mersenne Twister PRNG seeded with the CRC32 of the payload. In the analyzed sample, C2 reporting was configured to localhost, but the capability is fully implemented and operator-configurable. Following completion of encryption, if self-deletion is enabled in the configuration, Pay2Key overwrites its own binary with zeros using `dd` then securely wipes it using `shred -u -z -n 1`, eliminating forensic artifacts.

Mitigations

- **Restrict pvesh, qm, and pct to Authorized Accounts:** Pay2Key executes `pvesh get /cluster/resources, qm stop --skiplock, pct stop --skiplock,` and `pvesh delete` using root-level access. Limit the accounts that can invoke these commands through sudo policies. Alert on any non-administrative invocation of `pvesh delete` or use of the `--skiplock` flag outside maintenance windows. Proxmox does not enforce API-level access controls at the command line by default – this must be configured explicitly [M1026].
- **Monitor /etc/pve Access as a Detection Signal:** Any process accessing `/etc/pve` that is not a recognized Proxmox daemon or authorized administrative session should be treated as suspicious. This directory contains the cluster configuration and is not ordinarily accessed by user-space processes during normal operation. File integrity monitoring and process auditing on `/etc/pve` access patterns provides early warning of Pay2Key's environment detection phase [M1057].
- **Maintain Immutable Off-Host Backups:** Pay2Key's backup deletion routine specifically targets Proxmox-managed datastores via the cluster API, bypassing the `--protected` flag. Backups stored only within Proxmox-managed storage and relying on this flag for protection are vulnerable. Maintain at least one backup copy in immutable, off-host storage (object storage with object lock enabled, or an air-gapped system) that is not accessible via `pvesh` or any Proxmox node credential [M1053].
- **Alert on SELinux and AppArmor Disablement:** Pay2Key explicitly disables both SELinux (`setenforce 0`) and AppArmor (`aa-teardown, systemctl stop apparmor`) before beginning encryption. Configure alerting on any runtime invocation of `setenforce 0` or `aa-teardown` outside sanctioned change management windows. These commands are a reliable pre-encryption indicator in Pay2Key infections [M1054].
- **Audit Proxmox API and Task Logs:** Proxmox logs API calls through its task logging system. Review logs for unexpected calls to `/cluster/resources, /nodes/*/storage/*/content/*`, or stop operations outside maintenance windows. Automated anomaly

detection on `pvesh` usage patterns – particularly `pvesh delete` and `pvesh set ... --protected 0` – provides practical early warning of Pay2Key's backup destruction phase [M1057].

References

- *Halcyon / Beazley Security Labs, Pay2Key Iranian-Linked Ransomware is Back, Back Again (April 2026)*
- *Morphisec Threat Labs, Inside Pay2Key: Technical Analysis of a Linux Ransomware Variant (March 2026)*
- *FBI / CISA / DoD Cyber Crime Center, Joint Advisory on Iranian Cyber Actors (August 2024)*
- *KELA, Iranian Actors Target U.S. Critical Infrastructure Through Ransomware Proxies (March 2026)*

Source Summary

This Alert is based on Halcyon observations, open-source information, and ongoing research. Findings reflect our current understanding of threat actor activity and may be updated as new evidence emerges. Assessments may be revised as additional evidence becomes available.

The Halcyon Ransomware Research Center unites experts, drives smart policies, and delivers actionable intelligence to detect, disrupt, and defeat ransomware. Explore the Center's latest reports, analysis, and resources here.