

# **Cloud System Management, Patch Management, and Backup**

## **Week 8 Assignment**

Student: Shruti Malik

Course: ITEC5010 Security & Enterprise Networks

Date: August 27, 2025

## Introduction

Effective cloud system management requires two disciplines working in lockstep: rigorous patch management to reduce exploitable attack surface, and resilient data protection—including backup and tested recovery—to preserve availability and integrity during incidents. The sections below outline a router backup using Trivial File Transfer Protocol (TFTP), analyze tools that help ensure patching and backups occur reliably, highlight backup/restore best practices, note key considerations when deploying patches, and summarize disaster recovery (DR) best practices that also keep patch management on track.

### 1) Creating a Backup Using TFTP on a Router

Steps to back up a Cisco IOS router configuration to a TFTP server (use a dedicated management network because TFTP lacks encryption/authentication):

- Verify IP reachability from the router to the TFTP server (ping the server IP).
- Ensure a TFTP service is running on the server and note its TFTP root directory.
- Optionally set the router's TFTP source interface to the management interface: `ip tftp source-interface <interface>`.
- From privileged EXEC mode, run: `copy running-config tftp:`
- When prompted, enter the TFTP server IP and a destination filename (e.g., HQ-RTR-2025-08-27.cfg).
- Wait for transfer confirmation and verify the file in the TFTP root directory.
- To back up the startup config instead, use: `copy startup-config tftp:`. To back up the device image, use: `copy flash: tftp:`.

Pitfalls to avoid: (a) firewalls blocking UDP/69 or ephemeral UDP ports used by TFTP data channels; (b) incorrect TFTP root path or file permissions; (c) using TFTP over untrusted networks; (d) transferring large images over congested links without confirming bandwidth. When confidentiality or integrity is a concern, prefer SCP/SFTP over TFTP for configuration backups.

### 2) Tools That Ensure Effective Patching and Backups

Three widely used, cloud-grade tools help administrators verify that patching and backups are occurring as intended:

Tool	What it Ensures	Why it Matters
AWS Systems Manager Patch Manager	Automates OS/application patching across EC2, on-prem, and hybrid nodes; supports maintenance windows, patch baselines, and compliance	Centralized control and compliance evidence reduce mean-time-to-patch; pre/post scripts enable safe orchestration.

	dashboards.	
Azure Update Manager	Real-time or scheduled patching for Windows/Linux (Azure, Arc, on-prem); hotpatch options; assessments stored in Azure Resource Graph.	Consistent patch states at scale plus queryable evidence of installation and compliance.
AWS Backup / Azure Backup (Immutable Vaults)	Policy-driven snapshots/backups with cross-region/cross-account copies and immutability (Vault Lock / Immutable Vault).	Tamper-resistant backups withstand ransomware and operator error; enables provable restore readiness.

Tip: Integrate these with SIEM/SOAR so missed patches or failed backups trigger alerts and automated remediation (for example, retry or quarantine).

## 2a) How These Tools Integrate and Prove Compliance (Analysis)

Centralized evidence and dashboards: Patch Manager (AWS) and Update Manager (Azure) publish state and results to queryable control planes (AWS Systems Manager compliance views and Security Hub; Azure Resource Graph and Azure Policy). This enables a single-pane view across EC2/VMs, Azure VMs, Azure Arc-connected servers, and on-prem nodes, reducing audit toil and enabling exception workflows.

SIEM/SOAR pipelines: Forward patch and backup status to AWS Security Hub or Microsoft Sentinel to correlate signals (e.g., unmanaged/critical asset + CISA KEV CVE + missing patch) and trigger playbooks—isolating hosts, scheduling maintenance windows, or snapshotting before emergency patching.

Immutable backups vs ransomware: Vault-level immutability (AWS Backup Vault Lock; Azure Immutable Vaults) blocks common ransomware TTPs that try to modify or delete backups via stolen credentials or APIs. Retention locks enforce break-glass changes and provide assurance that restore points are trustworthy.

Closed-loop verification: Pair configuration/state management (SSM State Manager, Azure Policy/Update Manager) with backup job reports to enforce that pre-patch snapshots exist before production rollouts and that post-patch health checks pass—creating measurable SLOs for patch latency and restore readiness.

## 3) Three Best Practices for Backup and Restore

**Follow the 3-2-1-1-0 rule with immutability:** Maintain at least three copies on two media, one off-site, one immutable or air-gapped, and target zero errors verified by automated checks. Use cloud immutability features such as AWS Backup Vault Lock or Azure Immutable Vaults.

**Test restores regularly—don't just test backups:** Run scheduled DR drills and sandbox restores to verify Recovery Time Objective (RTO) and Recovery Point Objective (RPO). Document issues found and fix them within change windows.

**Make application-consistent backups with clear retention and classification:** Quiesce databases and apps to avoid corruption; document retention per data class (critical/regulated) and encrypt data in transit and at rest.

### 3a) Why These Practices Work (Analysis)

RTO/RPO alignment: Testing restores proves whether the organization can meet its RTO (how fast) and RPO (how much data loss). A green “backup success” alone is insufficient—drills surface issues like missing encryption keys, IAM permissions, or app version mismatches that would extend downtime.

Quiescing and application consistency: For databases and transactional systems, quiescing flushes in-memory transactions and pauses writes long enough to create a coherent snapshot. Examples: Windows VSS writers (e.g., SQL Server) and Linux pre/post scripts that place databases in backup mode (e.g., MySQL FLUSH TABLES WITH READ LOCK, Oracle BEGIN BACKUP), preventing torn pages and logical corruption upon recovery.

3-2-1-1-0 in practice: Keeping one immutable copy changes the attacker’s calculus—compromised credentials cannot shorten retention or delete restore points once locked. The “0” (zero errors) emphasizes monitoring backup logs, checksum/CRC verification, and sample restores so integrity is continuously proven.

Evidence and metrics: Track patch compliance %, time-to-patch for KEV-listed CVEs, backup success %, and—critically—restore success rate and MTTR from quarterly drills. Report these to governance alongside exceptions and risk acceptances.

## 4) Considerations When Deploying System Patches

- Staging & Testing: Validate in pre-production with representative workloads; then roll out in rings (pilot → broad).
- Maintenance Windows & Rollback: Define blackout periods, take pre-patch snapshots/backups, and maintain a tested rollback plan.
- Compatibility & Dependencies: Review app/database prerequisites; track drivers/firmware and container base image updates.
- Asset & SBOM Awareness: Map CVEs to assets/components (including third-party libraries) to understand true exposure.

- Network/Bandwidth: Throttle or cache updates at edge repositories to avoid saturating links.
- Evidence & Compliance: Capture patch status, exceptions/deferrals, and approvals for audits; prioritize KEV-listed CVEs.

## **5) Disaster Recovery Best Practices that Prevent Data Loss and Maintain Patch Management**

- Define RTO/RPO per workload via Business Impact Analysis; select DR strategy accordingly (backup/restore, pilot light, warm standby, active/active).
- Geo-separate and, where feasible, cross-account copies of backups/snapshots; enable immutability (Vault Lock / Immutable Vaults).
- Automate failover/fallback and runbooks with Infrastructure-as-Code; rehearse at least quarterly and capture evidence.
- Keep golden images and DR environments patched with the same baselines and compliance gates as production; rebuild images after critical CVEs.
- Integrate ransomware playbooks: maintain offline/imutable copies; validate clean-room recovery so malware is not reintroduced.
- Monitor KEV-listed vulnerabilities and expedite emergency patching across prod and DR to avoid vulnerable failover states.

## **Conclusion**

Patch management and data protection are mutually reinforcing. Automating patches with enterprise tools, designing backups for immutability and geographic separation, and continuously testing restores create a defensible posture against both opportunistic and targeted threats.

## **References (APA 7th)**

- Amazon Web Services. (2025). AWS Systems Manager Patch Manager.  
<https://docs.aws.amazon.com/systems-manager/latest/userguide/patch-manager.html>
- Amazon Web Services. (2024). Plan for disaster recovery (DR).  
<https://docs.aws.amazon.com/wellarchitected/latest/reliability-pillar/plan-for-disaster-recovery-dr.html>
- Amazon Web Services. (2024). AWS Backup Vault Lock. <https://docs.aws.amazon.com/aws-backup/latest/devguide/vault-lock.html>
- Amazon Web Services. (2025). AWS Security Hub.  
<https://docs.aws.amazon.com/securityhub/latest/userguide/what-is-securityhub.html>

Cisco. (2025). Understand how to backup and restore configuration files.  
<https://www.cisco.com/c/en/us/support/docs/ios-nx-os-software/ios-software-releases-122-mainline/46741-backup-config.html>

Cisco. (2023). Secure Copy (SCP).  
[https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst9500/software/release/17-13/configuration\\_guide/sys\\_mgmt/b\\_1713\\_sys\\_mgmt\\_9500\\_cg/secure\\_copy.html](https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst9500/software/release/17-13/configuration_guide/sys_mgmt/b_1713_sys_mgmt_9500_cg/secure_copy.html)

CISA. (2025). Known Exploited Vulnerabilities Catalog. <https://www.cisa.gov/known-exploited-vulnerabilities-catalog>

Microsoft. (2025). Azure Update Manager overview.  
<https://learn.microsoft.com/azure/update-manager/overview>

Microsoft. (2025). Concept of Immutable vault for Azure Backup.  
<https://learn.microsoft.com/azure/backup/backup-azure-immutable-vault-concept>

Microsoft. (2025). Microsoft Sentinel documentation.  
<https://learn.microsoft.com/azure/sentinel/>

National Institute of Standards and Technology. (2022). SP 800-40 Rev. 4: Guide to Enterprise Patch Management. <https://csrc.nist.gov/pubs/sp/800/40/r4/final>

National Institute of Standards and Technology. (2024). Cybersecurity Framework 2.0.  
<https://doi.org/10.6028/NIST.CSWP.29>

Veeam. (2024). 3-2-1-1-0 backup rule. <https://www.veeam.com/blog/321-backup-rule.html>