PATHSENTINEL / CYBERSECURITY RESEARCH

LESSONS LEARNED / IN-DEPTH DFIR REPORT

You Don't Need to Break Into the Operating System

ARMORY CASTLE, Two Zero-Days, and the Compromise of VMware's Control Plane

A technical investigation into a suspected China-linked cyberespionage operation, its attack chain, MITRE ATT&CK mapping, detection engineering, and incident-response lessons.

DOCUMENTED ATTACK PATH

BeyondTrust › Internal Linux › vCenter › ESXi › Virtual Disks

20 NOVEMBER 2026	ARMORY CASTLE / UNC5221	DFIR · CTI · SIGMA
Report source date	Case study	Detection engineering

English edition translated from the supplied Spanish PathSentinel report. Technical claims and citations are retained from the source document.

Contents

Technical case study · 20 chapters · 5 proposed Sigma rules · 9 technical references

01	Introduction: Why This Matters Even If You Don't Use VMware
02	Executive Summary
03	Understanding the Target: What VMware's Control Plane Actually Is
04	What Happened: Reconstructing the Incident Timeline
05	The First Zero-Day: BeyondTrust as the Entry Point
06	The Second Zero-Day: CVE-2026-123890 and the Breakdown of vCenter Authentication
07	From vCenter Control to Identity Forgery: Golden SAML
08	Observed TTPs: How ARMORY CASTLE Advanced
09	How Was the Intrusion Discovered?
10	MITRE ATT&CK Mapping: Demonstrated Techniques and Limits of the Evidence
11	Threat Hunting: What Should We Look For?
12	Proposed Sigma Rules for the SOC
13	Published Indicators of Compromise
14	Applying the Pyramid of Pain to ARMORY CASTLE
15	How Could the Attack Have Been Prevented?
16	Incident Response: What to Do If You Suspect vCenter Has Been Compromised
17	Lessons for AWS, Azure, Google Cloud, and Other Platforms
18	Lessons Learned: Seven Changes This Incident Should Drive
19	Recommended Action Plan for a Security Team
20	Conclusion: Protecting the Operating System Is No Longer Enough

1. Introduction: Why This Matters Even If You Don't Use VMware

Imagine an organization running fully patched Windows and Linux servers, endpoint detection and response (EDR) agents, multifactor authentication, network segmentation, and a security operations center continuously monitoring its systems.

Nevertheless, an attacker manages to extract sensitive information from a domain controller without having to execute malware inside its operating system.

How is this possible?

Because that server is not an isolated physical machine. It is a virtual machine whose disk, memory, configuration, and lifecycle depend on infrastructure managed from a higher layer.

An attacker does not need to defeat every defense within the guest operating system if they gain control of the platform administering it.

This is the problem illustrated by the investigation published on November 19, 2026, by Google Threat Intelligence Group (GTIG) and Mandiant.

According to the researchers, ARMORY CASTLE, a suspected China-linked cyberespionage actor previously tracked as UNC5221, chained two zero-day vulnerabilities to move from a compromised remote-support solution into an organization's internal environment and ultimately into the VMware vSphere control plane.

Once there, the actor used purpose-built tools to compromise vCenter authentication, obtain credentials, access ESXi hypervisors, and extract files directly from virtual disks.

The importance of this case extends well beyond VMware.

Virtually every modern architecture includes some form of **control plane**: the layer used to create, manage, connect, modify, and delete computing resources.

That might be VMware vCenter, a hypervisor-management platform, a cloud infrastructure console, Kubernetes, a backup platform, or an orchestration system.

The principle is the same:

> **When an attacker controls the infrastructure that governs systems, defenses installed inside those systems may no longer be sufficient.**

This case therefore matters to security leaders, incident response teams, threat intelligence analysts, systems administrators, cloud architects, and organizations that do not use VMware at all.

This is not another ransomware incident in which virtual machines are encrypted. It is an espionage operation aimed at obtaining sensitive information by taking advantage of a management layer that many organizations do not monitor as comprehensively as their servers.

Above all, it forces an uncomfortable question:

Are we protecting our servers better than the systems capable of controlling them?

2. Executive Summary

On November 19, 2026, Mandiant and GTIG documented active exploitation of **CVE-2026-123890**, a critical authentication vulnerability in VMware vCenter Server.

Broadcom assigned it a **CVSS v3.1 score of 9.8/10** and released security updates through advisory **VMSA-2026-0009.1**.

The investigation links the exploitation to ARMORY CASTLE, an actor suspected of operating within the People's Republic of China's cyberespionage ecosystem.

The principal findings are summarized below.

Item	Finding
Actor	ARMORY CASTLE, formerly UNC5221
Attribution	Suspected China link, according to GTIG
Objective	Espionage and access to high-value information
First zero-day	CVE-2026-102764, BeyondTrust Remote Support
Second zero-day	CVE-2026-123890, VMware vCenter Server
Second CVE severity	Critical, CVSS 9.8
Exploitation tool	BONESAW
Identity-forgery tool	SAWSNAKE
Persistence and access	Web shells, backdoors, SSO accounts, and SSH keys
Observed impact	vCenter compromise, ESXi access, and file extraction from virtual disks

The central finding is that **vCenter exploitation was not the initial entry point**. It occurred later in a multi-stage lateral-movement operation.

This distinction matters: an organization can keep vCenter inaccessible from the public internet and still be exposed if another compromised internal system can reach its management interfaces.

3. Understanding the Target: What VMware's Control Plane Actually Is

Before examining the vulnerability, three components need to be distinguished.

3.1. ESXi: The Hypervisor

VMware ESXi is the hypervisor that allows multiple virtual machines to run on a single physical server.

It manages resources such as CPU, memory, storage, and virtual networking.

Its position in the architecture gives it capabilities over guest systems that do not necessarily depend on the Windows or Linux credentials used within each virtual machine.

3.2. vCenter Server: Centralized Management

vCenter is the system through which multiple ESXi hypervisors and their resources are administered.

It supports inventory management, permissions, configurations, virtual machines, storage operations, and other infrastructure components.

In many organizations it is deployed as **vCenter Server Appliance (VCSA)**, a virtual appliance based on Photon OS.

3.3. Virtual Machines: The Guest Systems

These are the Windows and Linux servers on which organizations typically install antivirus, EDR agents, monitoring agents, and other security tools.

But there is a fundamental difference between controlling a guest operating system and controlling the infrastructure from which that guest is managed.

An administrator with sufficient vSphere privileges can perform operations on virtual machines without necessarily authenticating to their operating systems.

That does not mean every vCenter compromise automatically gives unrestricted access to all data on every VM. What is possible depends on the privileges obtained, configuration, storage, and security protections in place.

Nevertheless, ARMORY CASTLE demonstrates that, under certain conditions, an attacker can **extract data directly from virtual storage**, bypassing the traditional guest-OS access paths.

That shift in perspective should inform defensive strategy.

4. What Happened: Reconstructing the Incident Timeline

Mandiant's investigation allows us to reconstruct an attack chain that begins outside VMware and ends with sensitive information being extracted from a hypervisor.

Documented Timeline

Date	Event
July 13, 2026	A compilation timestamp recovered from BONESAW. This is a clue, not proof of its actual development date.
July 29, 2026	Disclosure of CVE-2026-59309, a separate vCenter vulnerability.
Late August 2026	Exploitation of CVE-2026-102764 in BeyondTrust Remote Support, according to Mandiant.
	Evidence that BONESAW was executed against vCenter using CVE-2026-123890.

Date	Event
Early September 2026	
October 15, 2026	GTIG and Mandiant publish the BeyondTrust investigation.
November 19, 2026	Public disclosure of CVE-2026-123890 and publication of the ARMORY CASTLE/vSphere investigation.
November 20, 2026	Broadcom's advisory is updated to VMSA-2026-0009.1.

Important qualification: The BONESAW compilation timestamp could have been manipulated and does not establish when the tool was developed. The exploitation evidence primarily comes from forensic reconstruction of the incidents.

The End-to-End Attack Chain

1. **Initial access:** Exploitation of BeyondTrust Remote Support.
2. **Persistence:** Installation of malicious tools on the compromised appliance.
3. **Lateral movement:** Access to an internal Linux appliance with less comprehensive security coverage.
4. **Privilege escalation:** Acquisition of root-level capabilities.
5. **Reconnaissance:** Identification of VMware services and interfaces.
6. **vCenter exploitation:** Deployment of BONESAW against CVE-2026-123890.
7. **Identity compromise:** Extraction of secrets, creation of SSO accounts, and forgery of SAML tokens.
8. **ESXi access:** Use of administrative credentials and establishment of SSH-based persistence.
9. **Data access:** Exploration of VMFS volumes and extraction of files from virtual disks.
10. **Exfiltration and additional persistence:** Preparation of files, movement between compromised hosts, and deployment of backdoors.

The distinctive feature is the chaining of two zero-days with tools specifically developed for different platforms.

This was not a single exploit fired at a vulnerable server. It was a multi-stage operation.

5. The First Zero-Day: BeyondTrust as the Entry Point

The first documented vulnerability was **CVE-2026-102764**, affecting BeyondTrust Remote Support.

This product is used to provide technical support and administer devices remotely. By design, it may have exceptional connectivity and privileged access inside an organization.

Mandiant observed ARMORY CASTLE exploiting the vulnerability before public disclosure.

The flaw allowed unauthorized file-upload operations and could facilitate control of the affected appliance.

The earlier GTIG investigation identified tools including:

- **SAWFLY:** A web shell used to maintain access.
- **BARECLAW:** A tool facilitating communications and lateral movement.

Persistence and evidence-removal techniques were also observed.

The flaw was addressed through **BT26-05**, according to the vendor.

First Lesson: The Risk of Privileged-Access Systems

Remote-support tools, VPNs, identity-management systems, backup administration platforms, and other infrastructure appliances have something in common: they frequently enjoy exceptional network connectivity.

That is precisely why they are attractive to attackers.

The problem is not merely that they might contain a vulnerability. Once compromised, they can become ideal launch points from which to reach systems that were never exposed directly to the internet.

A compromised management system can inherit access opportunities unavailable to a conventional user workstation.

6. The Second Zero-Day: CVE-2026-123890 and the Breakdown of vCenter Authentication

This is the technical core of the investigation.

6.1. What Vulnerability Was Exploited?

CVE-2026-123890 is a critical **authentication-bypass** vulnerability in VMware Directory Service.

According to Mandiant, it affects the **Secure Remote Password (SRP-6a)** authentication mechanism implemented over **DCE/RPC** interfaces.

The exploit involves the `rpc_srp_verifier` service, accessible over **TCP/2012**.

Under normal conditions, SRP is intended to support secure authentication without sending the password itself over the network.

However, the vulnerable implementation had cryptographic and authorization weaknesses that made it possible to recover information related to the session key.

Mandiant also describes exploiting predictable properties of UUIDv1 generation.

The consequence was particularly serious:

> **An attacker with network access to the vulnerable service could establish a session authenticated as `administrator@vsphere.local` without knowing that account's password.**

6.2. Why the Flaw Was Critical

The problem was not limited to the disclosure of configuration information.

The resulting session could be used for privileged operations against vCenter directory services.

The investigation documents abuse of the `vmdirdbcp` RPC interface to retrieve files, including the database:

`/storage/db/vmware-vmdir/data.mdb`

This database is a high-value target because it contains vCenter directory information and material associated with its identity mechanisms.

Once obtained, an attacker could extract secrets and prepare additional fraudulent authentication activity.

6.3. BONESAW: The Exploitation Tool

Mandiant identified a Go-based utility named **BONESAW**, which automated several stages of the compromise.

Its documented capabilities included:

1. Exploiting CVE-2026-123890.
2. Obtaining a copy of the vCenter LMDB database.
3. Extracting cryptographic material associated with identity services.
4. Creating administrative SSO accounts.
5. Facilitating compromise of vCenter authentication and authorization mechanisms.

Analysis of the recovered binary revealed the internal name `vcenter-db-teamcheck-go` and compilation artifacts consistent with a Windows development environment.

These findings help characterize the tool, but are not enough to identify its developers.

6.4. Distinguishing CVE-2026-123890 from CVE-2026-59309

It is important not to confuse **CVE-2026-123890** with **CVE-2026-59309**.

Both affect vCenter authentication mechanisms, but they are distinct vulnerabilities.

CVE-2026-59309 affects an LDAP path within the `vmdir` service, associated with TCP/389 and TCP/636.

CVE-2026-123890 affects a DCE/RPC interface related to SRP on TCP/2012.

Mandiant documented BONESAW running against a server that had already been patched against CVE-2026-59309.

This is important: fixing the earlier flaw did not resolve the second one.

6.5. Affected Versions and Updates

Broadcom's official advisory specifies the following corrected versions:

Product or release branch	Corrected version / action
vCenter 9.1.x	9.1.1.0200
vCenter 9.0.x	9.0.2.0200
vCenter 8.0	8.0 U3m
vCenter 7.0	Contact Broadcom if covered by an extended support contract
VMware Cloud Foundation 5.x	5.2.5

Additional guidance is available for other solutions that integrate vCenter.

Broadcom did not publish an official workaround for this CVE. Its recommended remediation is to install the applicable security update.

Restricting connectivity to vulnerable services may reduce exposure, but does not replace patching.

Advisory VMSA-2026-0009.1 also covers **CVE-2026-123891, CVE-2026-123892, CVE-2026-123893, and CVE-2026-123894** in components such as SDDC Manager and VMware Tools.

Those additional flaws should not be presented as having been exploited by ARMORY CASTLE. The documented exploitation in this operation relates to **CVE-2026-123890** and, earlier, **CVE-2026-102764**.

7. From vCenter Control to Identity Forgery: Golden SAML

One of the most revealing stages of this case is the deployment of **SAWSNAKE**.

This tool, also written in Go, could process cryptographic material belonging to vCenter identity services. Mandiant associates it with **Golden SAML** techniques.

What Is Golden SAML?

SAML is a standard for exchanging authentication information between systems.

In a legitimate process, an identity provider issues a signed assertion that another service accepts as proof of authentication.

If an attacker obtains the private key used to sign those assertions, they may be able to create tokens that appear to come from a legitimate authority.

The problem is no longer just a stolen password. The attacker may be able to manufacture apparently valid proof of identity.

What Did SAWSNAKE Do?

According to the investigation, SAWSNAKE could:

- Analyze the `data.mdb` database.
- Extract private keys and trusted certificates.

- Generate signed SAML assertions.
- Obtain administrative session cookies.
- Impersonate identities when communicating with vCenter.

This has an important implication for MFA-based strategies.

Multifactor authentication cannot protect against every scenario if an attacker compromises the authority responsible for issuing or validating authentication tokens.

That does not make MFA useless. Rather, it means that protecting signing keys, identity services, and trust relationships is just as important as authenticating users.

Lesson for the SOC

A technically valid authentication event is not necessarily legitimate.

When compromise of a signing key is suspected, reviewing failed logins is not enough. Investigators must examine successful sessions, newly created identities, privilege changes, and the integrity of the trust infrastructure.

8. Observed TTPs: How ARMORY CASTLE Advanced

8.1. Infrastructure Reconnaissance

After compromising the internal Linux appliance, the actor conducted network scans with `nmap`.

Mandiant observed probing of ports including:

- **TCP/443:** Web interfaces and APIs.
- **TCP/2012:** The RPC service relevant to the exploit.
- **TCP/389:** LDAP.
- **TCP/5480:** VCSA management.
- **TCP/9080:** Another service included in the observed reconnaissance.

The actor also investigated paths such as `/sdk` and `/ui`.

This behavior indicates targeted identification of VMware systems.

Defensive opportunity: Connections from a support appliance or a non-administrative Linux host to a vCenter server's TCP/2012 should be regarded as anomalous unless operationally justified.

8.2. Web Shells on the Linux Appliance

Mandiant identified two tools:

- **SLIMSAVAGE**, deployed as `ActiveSer.jsp`, enabled remote command execution.
- **CAGEMATCH**, deployed as `u.jsp`, enabled file uploads and staging of additional tools.

The combination is important: one web shell executed commands while the other facilitated moving binaries into place.

8.3. Privilege Escalation

The actor used `gdb` to modify permissions associated with `/bin/env`, enabling a privileged execution path through the **SUID** bit.

This technique deserves attention because it does not rely on a conventional kernel vulnerability. Instead, it abuses legitimate system tools to change security-relevant behavior.

Threat hunting should therefore cover both abnormal debugger execution and permission changes to sensitive binaries.

8.4. vCenter Compromise

After preparing the environment, the actor used BONESAW against vCenter.

Mandiant observed the creation of an administrative SSO account and extraction of the LMDB database.

SAWSNAKE was subsequently deployed to facilitate SAML token forgery.

8.5. Backdoors and Persistence

The investigation also identifies **BEARBONES**, a backdoor based on a modified version of Dropbear SSH.

Its characteristics include changes to the authentication mechanism and the ability to log credentials.

A particularly distinctive indicator is its SSH banner:

```
SSH-2.0-pulsesecure_23345
```

Mandiant observed attempts to listen on **TCP/53** and **TCP/33992**.

The researchers also identified **LAZYBONES**, a Go-based backdoor that used **Slack** as a command-and-control channel.

Its communications were carried over HTTPS, using encrypted and encoded messages.

Using a legitimate service such as Slack complicates strategies based exclusively on blocking suspicious domains.

8.6. Temporarily Enabling SSH on vCenter

ARMORY CASTLE used the VCSA management API to enable SSH.

The investigation describes an HTTP `PUT` request to:

```
/rest/appliance/access/ssh
```

After accessing the appliance and retrieving sensitive information, the actor disabled the service again.

This provides an especially useful detection lesson:

> **It is not enough to check whether SSH is enabled now. You need to record when it was enabled, from where, by whom, and for how long it remained available.**

8.7. ESXi Access and Persistence Through SSH Keys

Mandiant observed authentication attempts to ESXi using `vpxuser` and `root`.

The actor also added a public key to the root user's authorized-keys file.

This allows SSH persistence without relying on the original password.

8.8. The Decisive Phase: Extracting Data from Virtual Disks

The investigation documents use of a tool named `vmfs_browser.py`.

The attacker used it to explore VMFS volumes and VMDK virtual-disk files.

Observed targets included files associated with a Windows domain controller:

- `NTDS`
- `SYSTEM`
- `SECURITY`

These files are particularly sensitive because they may contain information essential to Active Directory security.

The actor prepared compressed archives for subsequent transfer.

This is where the article's central idea becomes concrete: The attacker accessed guest-system information from the virtual-storage layer, without needing to run the extraction tool inside that Windows instance.

An EDR installed on the domain controller may not see a disk read performed from the hypervisor as though it were a process running inside the guest.

This does **not** mean the operation was undetectable. It means the sensors must also cover the virtualization infrastructure and its administrative paths.

9. How Was the Intrusion Discovered?

It is essential to distinguish what is known from what is not.

Documented: Mandiant reconstructed the operation through incident-response work and forensic evidence, including recovered logs, artifacts, and tools.

Not publicly established: The exact first alert or event that prompted the affected organization to request assistance.

It would be unsound to claim the breach was initially discovered by an EDR, a SIEM, a particular vCenter alert, or proactive hunting when the source does not specify this.

Evidence Used to Reconstruct the Case

The investigation describes several types of artifacts:

On BeyondTrust: Access logs, syslog, other internal application files, evidence of modifications, and deployed tools.

On the Linux appliance: JSP web shells, malicious binaries, download operations, executed commands, and evidence associated with privilege escalation.

On vCenter: SSO account creation, use of administrative interfaces, SSH enablement, and access to sensitive information.

On ESXi: Authentication events, SSH keys, volume-exploration activity, and operations on virtual-machine files.

In network infrastructure: Reconnaissance, host-to-host connections, and communications associated with persistence tools.

The Forensic Challenge

In an investigation of this kind, guest-operating-system logs are insufficient.

The timeline needs to correlate:

1. The initially compromised appliance.
2. Intermediate systems used for lateral movement.
3. VMware's management plane.
4. ESXi hypervisors.
5. Network logs.
6. Identity systems.
7. Evidence of transfer and exfiltration.

If one of these layers does not retain logs, reconstructing the incident may be incomplete.

10. MITRE ATT&CK Mapping: Demonstrated Techniques and Limits of the Evidence

The following mapping is a defensive interpretation of the behavior described by Mandiant. **It is not an official ATT&CK matrix published by GTIG for this incident.**

ATT&CK technique	ID	Evidence in this case
Exploit Public-Facing Application	T1190	Initial exploitation of BeyondTrust Remote Support
Network Service Discovery	T1046	`nmap` scans of VMware services
Server Software Component: Web Shell	T1505.003	SLIMSAVAGE and CAGEMATCH

ATT&CK technique	ID	Evidence in this case
Unix Shell	T1059.004	Commands executed through a web shell
Setuid and Setgid	T1548.001	Manipulation of SUID permissions
Exploitation of Remote Services	T1210	vCenter exploitation from an internal system
Create Account	T1136	Creation of an administrative SSO account
Forge Web Credentials: SAML Tokens	T1606.002	Golden SAML capabilities of SAWSNAKE
Remote Services: SSH	T1021.004	Lateral access through SSH
Account Manipulation: SSH Authorized Keys	T1098.004	Addition of a public key for root
OS Credential Dumping: NTDS	T1003.003	Extraction of NTDS files from a virtual disk
Archive Collected Data	T1560	Preparation of compressed archives
Web Service	T1102	Slack used as C2 infrastructure by LAZYBONES
Create or Modify System Process: Systemd Service	T1543.002	Persistence through a systemd service
Indicator Removal: File Deletion	T1070.004	Deletion of tools and artifacts

Three confidence levels must be kept separate:

- **Observed:** Operations explicitly described by Mandiant.
- **Inferred through ATT&CK:** The correspondence between those operations and framework techniques.
- **Not demonstrated:** Actions that may have been technically possible given the privileges acquired, but whose execution was not documented.

For example, administrative control of vCenter can facilitate destructive operations. However, that does not demonstrate that ARMORY CASTLE deployed ransomware or destroyed virtual machines in this incident.

Threat intelligence loses value when it turns technical possibilities into established facts.

11. Threat Hunting: What Should We Look For?

Effective hunting must go beyond checking published indicators. We need to investigate transitions between attack stages.

Hypothesis H1: Unexpected Access to Management Services

Hypothesis: A system outside the management network attempts to contact sensitive vCenter interfaces.

Telemetry: Firewall, NetFlow, Zeek, IDS/NDR.

Hunt: Connections to vCenter over TCP/2012, 5480, 22, 389, and 636 from unauthorized systems.

Prioritization: Traffic from support appliances, VPN appliances, application servers, or user endpoints.

Hypothesis H2: Unauthorized SSO Identity Creation

Hypothesis: The attacker creates an administrative account to maintain access.

Telemetry: vCenter identity and SSO audit events.

Hunt: User creation, assignment of global privileges, and authentication from anomalous source systems.

Prioritization: Accounts created outside approved change windows or used immediately after creation.

Hypothesis H3: Temporary SSH Activation

Hypothesis: An attacker enables SSH on VCSA to extract secrets, then turns it off.

Telemetry: VAMI logs, configuration events, authentication logs, and network traffic.

Hunt: Requests to `/rest/appliance/access/ssh`, service-state changes, and subsequent SSH connections.

Prioritization: Enable/disable cycles over a short period without an approved change.

Hypothesis H4: Access to Virtual Storage from the Hypervisor

Hypothesis: An attacker explores VMDK files and extracts sensitive data from ESXi.

Telemetry: ESXi logs, SSH sessions, SCP transfers, storage operations, and network logs.

Hunt: Unusual administrative access, use of VMFS-browsing tools, compressed archives in datastores, and transfers originating from the hypervisor.

Prioritization: Operations targeting virtual disks belonging to domain controllers, identity servers, and other critical assets.

Hypothesis H5: Persistence on Intermediate Linux Systems

Hypothesis: The attacker uses web shells, debugging tools, and binaries stored in temporary directories.

Telemetry: auditd, Linux EDR, file integrity monitoring, syslog, and web-server logs.

Hunt: Unexpected JSP files, shell processes launched by Java, `gdb` operating on privileged processes, and SUID permission changes.

12. Proposed Sigma Rules for the SOC

The following rules are **original detection proposals based on the documented TTPs**. They are **not official Mandiant rules**, nor have they been validated in a production environment.

Sigma is a detection format, not a telemetry source. Each rule therefore needs an actual data source and, in some cases, field mapping before translation into the syntax of a particular SIEM.

Rule 1. Detection of Web Shells Associated with the Campaign

Goal: Identify creation of filenames used by SLIMSAVAGE and CAGEMATCH.

Required source: File-creation events from FIM, enriched auditd, or Linux EDR.

```
title: ARMORY CASTLE - Known JSP Webshell File Creation
id: 7d26d2f1-9725-40e9-98f0-7a93416b2031
status: experimental
description: Detects filenames associated with SLIMSAVAGE and CAGEMATCH.
references:
  - https://cloud.google.com/blog/topics/threat-intelligence/armory-castle-exploits-vsphere
logsource:
  category: file_event
  product: linux
detection:
  selection:
    TargetFilename|endswith:
      - '/ActiveSer.jsp'
      - '/u.jsp'
  condition: selection
falsepositives:
  - Legitimate application files with matching names
level: high
tags:
  - attack.persistence
  - attack.t1505.003
```

Limitation: Filenames are easy to change. This can help find known artifacts but is not sufficient protection against new variants.

Rule 2. Suspicious GDB Execution Using Temporary Command Files

Goal: Detect activity similar to the technique used to modify privileged permissions.

Required source: Linux process-creation telemetry.

```
title: ARMORY CASTLE - Suspicious GDB Command File Execution
id: 541bb99c-8304-4e35-bfe2-c4cb5fb65190
status: experimental
description: Detects GDB execution using command files staged in temporary directories.
references:
  - https://cloud.google.com/blog/topics/threat-intelligence/armory-castle-exploits-vsphere
logsource:
  category: process_creation
  product: linux
detection:
  selection_image:
    Image|endswith:
      - '/gdb'
  selection_command:
    CommandLine|contains|all:
      - '--command='
      - '/tmp/'
  condition: selection_image and selection_command
falsepositives:
  - Authorized debugging or maintenance activities
level: high
tags:
  - attack.privilege-escalation
  - attack.t1548.001
```

Limitation: This detects a particular implementation of the technique. It will not detect every SUID permission change or every malicious use of GDB.

It should be supplemented by auditing permission changes on sensitive binaries.

Rule 3. SSH Management Through the VCSA API

Goal: Alert on use of the administrative interface observed during the intrusion.

Required source: VAMI HTTP logs exposing method and path.

The `cs-method` and `cs-uri-stem` fields reflect a proposed normalized HTTP schema; they must be mapped from the actual VAMI log format.

```
title: VMware VCSA - SSH Enablement Through Management API
id: e8ac50d5-287f-45be-a1ca-8fb09c78ac34
status: experimental
description: Detects PUT requests to the VCSA SSH management API.
references:
  - https://cloud.google.com/blog/topics/threat-intelligence/armory-castle-exploits-vsphere
logsource:
  category: webserver
detection:
  selection:
    cs-method: 'PUT'
    cs-uri-stem: '/rest/appliance/access/ssh'
  condition: selection
falsepositives:
  - Authorized maintenance enabling or disabling SSH
level: high
tags:
  - attack.lateral-movement
  - attack.t1021.004
```

Recommendation: Correlate the request with the authenticated user, source IP, SSH state before and after the request, and subsequent connections to port 22.

The `PUT` request identifies a configuration change, but does not prove that SSH was enabled; determining that requires the request body or an observed state change.

Rule 4. Access to TCP/2012, a Sensitive vCenter Interface

Goal: Identify connections to the service associated with CVE-2026-123890.

Required source: Normalized connection events from a firewall or NDR platform.

```
title: VMware vCenter - Connection to Sensitive RPC Service
id: 04f5df90-a3e4-4d67-93bc-56ccfc0474b8
status: experimental
description: Identifies connections to TCP port 2012 for further investigation.
references:
  - https://cloud.google.com/blog/topics/threat-intelligence/armory-castle-exploits-vsphere
logsource:
  category: network_connection
detection:
  selection:
    DestinationPort: 2012
  condition: selection
falsepositives:
  - Legitimate VMware infrastructure communication
level: low
tags:
  - attack.discovery
  - attack.t1046
```

Important: This is deliberately an exposure-hunting query, not proof of exploitation.

In production, it should be scoped to inventoried vCenter destinations, with approved sources excluded.

A connection to TCP/2012 alone does **not** demonstrate use of BONESAW or CVE-2026-123890.

Rule 5. SSH Banner Associated with BEARBONES

Goal: Identify a comparatively specific network artifact.

Required source: Zeek SSH logs in which the server banner is available in the `server` field.

```
title: ARMORY CASTLE - BEARBONES Suspicious SSH Server Banner
id: 41a92b1f-6712-4b78-b56a-9f7d96bc9d54
status: experimental
description: Detects the SSH banner associated with BEARBONES.
references:
  - https://cloud.google.com/blog/topics/threat-intelligence/armory-castle-exploits-vsphere
logsource:
  product: zeek
  service: ssh
detection:
  selection:
    server|contains: 'pulsesecure_23345'
  condition: selection
falsepositives:
  - Deliberate banner emulation
level: high
tags:
  - attack.command-and-control
```

Limitation: The precise field depends on the Zeek version and log normalization, and the SSH banner can be changed by the attacker.

Which Rules Would I Prioritize?

For a SOC applying the lessons of this case, the priority should not be limited to rules based on malware names.

More valuable detections should cover:

- Unauthorized access to management services.
- Creation of privileged vCenter accounts.
- SSH enablement and configuration changes.
- Manipulation of SSH keys.
- Anomalous activity involving virtual disks.
- Extraction of data from infrastructure systems.
- Changes to identity mechanisms.

The Sigma rules above provide a starting point. Behavioral detections should be expanded with environment-specific correlations.

13. Published Indicators of Compromise

GTIG included the following network indicators associated with suspected obfuscation infrastructure used by the actor:

```
103.147.251.5
103.190.38.3
118.174.128.34
118.174.128.36
118.174.128.38
176.9.89.66
185.71.15.109
220.121.128.41
222.108.77.44
61.216.56.127
81.45.89.43
```

The researchers also published the following filenames and paths:

Indicator	Associated tool
`ActiveSer.jsp`	SLIMSAVAGE
`u.jsp`	CAGEMATCH
`/tmp/123`	BONESAW
`/tmp/sss`	SAWSNAKE
`/tmp/dbr`	BEARBONES
`/usr/bin/vmtoolscs`	LAZYBONES

Other relevant indicators include:

- SSH banner `SSH-2.0-pulsesecure_23345`.
- Unauthorized creation of administrative SSO accounts.
- Abnormal use of `/rest/appliance/access/ssh`.
- Unauthorized SSH keys in privileged accounts.
- Temporary and hidden directories used to stage tools and archives.

CTI warning: Published IP addresses should not be treated as permanently malicious. They may refer to third-party infrastructure that was itself compromised and whose owner or purpose may change.

Use them for contextualized historical hunting, not as standalone proof of attribution.

GTIG also published YARA rules specific to SLIMSAVAGE, CAGEMATCH, BONESAW, SAWSNAKE, BEARBONES, and LAZYBONES.

To preserve their original conditions and avoid transcription errors, obtain those rules directly from the GTIG report linked in the bibliography.

14. Applying the Pyramid of Pain to ARMORY CASTLE

David Bianco's **Pyramid of Pain** describes the effort an adversary must expend when defenders identify and block different types of indicators.

The higher a detection sits on the pyramid, the harder it generally is for the attacker to adapt without changing how they operate.

Level 1. Hashes

Example: Hashes of individual BONESAW or SAWSNAKE samples.

Pain for the attacker: Low.

Recompiling a tool can generate a different hash.

Defensive value: Forensic identification and hunting for known samples.

Level 2. IP Addresses

Example: The eleven addresses published by GTIG.

Pain: Relatively low.

An actor can switch to new addresses, proxies, or compromised devices.

Defensive value: Historical correlation, enrichment, and temporary blocking.

Level 3. Domains and Communication Services

Example: C2 infrastructure and legitimate services abused to transport commands.

Pain: Variable.

LAZYBONES illustrates why blocking an entire domain such as Slack may not be practical.

Defensive value: Analysis of destinations, communication patterns, and unauthorized accounts or applications.

Level 4. Network and Host Artifacts

Examples:

- SSH banner `pulsesecure_23345`.
- Unexpected JSP files.
- Use of temporary directories for tool staging.
- SSH keys added to privileged accounts.

Pain: Medium.

An attacker can change filenames and strings, but doing so requires adaptations to tools and procedures.

Level 5. Tools

Examples:

- BONESAW.
- SAWSNAKE.
- BEARBONES.
- LAZYBONES.
- VMFS exploration utilities.

Pain: High.

Robustly detecting these tools' capabilities and behavior may force an actor to redesign parts of the operation.

Level 6. Tactics, Techniques, and Procedures

Examples:

- Pivoting from a compromised appliance into the control plane.
- Exploiting infrastructure authentication services.
- Creating privileged SSO accounts.
- Obtaining signing keys.
- Temporarily enabling SSH.
- Reading virtual disks to extract sensitive information.
- Maintaining persistence through SSH keys and legitimate services.

Pain: Very high.

If an organization detects these behaviors regardless of the tool used, the attacker must change their operational strategy.

The Pyramid's Central Lesson

Defensive priorities should move from:

“Do we know ARMORY CASTLE's IP addresses and hashes?”

To:

“Would we detect any actor attempting to compromise our control plane and read virtual disks through it?”

The second question is far more useful.

15. How Could the Attack Have Been Prevented?

No single measure guarantees prevention of every intrusion, particularly when zero-days are involved.

Nevertheless, this incident reveals multiple points at which defenders might have broken the attack chain.

15.1. Treat vCenter as a Highest-Criticality Asset

In organizations using Active Directory, vCenter can have an impact comparable to other **Tier 0** assets.

Its compromise may enable operations against servers containing critical information.

Its administration should therefore be subject to exceptionally strict controls.

15.2. Segment the Management Plane

Keeping vCenter off the public internet is not enough.

Organizations must also restrict which internal systems can reach its interfaces.

A compromised remote-support appliance should not automatically have connectivity to vCenter identity and management services.

As dictated by legitimate operational requirements, policies should restrict access to ports including **TCP/2012, 443, 5480, and 22**.

Segmentation should rely on independent and verifiable controls, ideally located outside the system being protected.

15.3. Do Not Rely Exclusively on the VCSA Firewall

Mandiant warns of limitations in the firewall administered through VAMI.

The report highlights issues concerning granularity, visibility, and reliance on the administrative interface itself.

It recommends strengthening network controls and recording rejected connection attempts.

GTIG also links Mandiant's **VCSA Hardening & Logging Tool**, which helps configure restrictive policies and additional auditing.

However, directly modifying VCSA `iptables` rules can disrupt connectivity and create support-compatibility concerns.

Do not deploy such modifications to production without laboratory testing, dependency review, change control, and a console-based recovery procedure.

15.4. Use Privileged Access Workstations

vCenter and ESXi administration should be performed from controlled management workstations using separate identities and restricted access.

Allowing an administrative account to be used from arbitrary computers on the network expands the attack surface.

15.5. Centralize Logs Before an Incident

It is essential to forward VCSA and ESXi logs to a protected external repository.

It is equally important to verify that operating-system audit events from the appliance are actually being forwarded.

Mandiant notes that the usual VCSA syslog configuration does not necessarily ensure remote forwarding of all `auditd` events.

This matters because an attacker with privileged access may delete local logs.

15.6. Monitor Integrity

Recommended monitoring targets include:

- Critical configuration files.
- SSH keys.
- systemd services.
- SUID permission changes.
- Unexpected binaries.
- Changes to authentication mechanisms.
- New accounts and privilege assignments.

15.7. Restrict Outbound Network Traffic

Infrastructure appliances should not have unrestricted internet access.

An egress policy based on approved destinations and services can make C2 communication and exfiltration more difficult.

15.8. Patch — But Do Not Depend on Patching Alone

Installing official updates is essential.

Yet during the period when a vulnerability is unknown, no patch is available.

Defensive strategy must therefore assume that another zero-day may emerge.

Segmentation, least privilege, monitoring, and protection of identity services are precisely the measures that reduce the impact of such a scenario.

16. Incident Response: What to Do If You Suspect vCenter Has Been Compromised

When there is a well-founded suspicion of compromise, the response should not stop at upgrading vCenter and changing the administrator password.

This case shows that attackers may have obtained keys, tokens, accounts, and access to dependent systems.

Phase 1. Containment

Restrict connections to the management plane and stop unauthorized access.

Before powering off systems or indiscriminately changing services, assess whether volatile evidence must be preserved.

The response strategy must be adapted to incident severity and business-continuity requirements.

Phase 2. Forensic Preservation

Collect evidence using documented procedures:

- VCSA logs.
- ESXi logs.
- SSO events.
- Administrative-operation history.
- VAMI logs.
- SSH logs.
- Network telemetry.
- Evidence of account and permission changes.
- Suspicious artifacts on intermediate Linux systems.

Where possible, preserve verifiable copies and maintain chain of custody.

Phase 3. Identity Analysis

Investigate:

- Recently created SSO accounts.
- Assigned global privileges.
- Anomalous administrative sessions.
- Integrity of STS signing keys.
- Credentials and secrets accessible from vCenter.

If evidence suggests signing keys were extracted, recovery should include their replacement using vendor-compatible procedures and invalidation of affected trust relationships.

Changing a password does not necessarily neutralize a stolen signing key.

Phase 4. ESXi Analysis

Review:

- `root` and `vpxuser` access.

- Authorized SSH keys.
- SCP sessions.
- Changes to management services.
- Unexpected files in datastores.
- Virtual-disk exploration and copy operations.

Phase 5. Scope Assessment

If the extraction of domain-controller files is confirmed, expand the investigation to Active Directory.

A broader identity-recovery process, including controlled rotation of domain secrets, may be necessary.

These operations should take place after containment is secured and under specific procedures to avoid further compromise opportunities or unnecessary disruption.

Phase 6. Recovery

Restore compromised components from a trusted state.

Apply Broadcom security updates.

Remove unauthorized persistence, accounts, and keys.

Review credentials that may have been exposed.

Confirm recovered systems no longer trust compromised cryptographic material.

Phase 7. Follow-Up Monitoring

Maintain enhanced monitoring of authentication, management access, and connections between network segments.

An espionage operation may retain multiple persistence mechanisms. Recovering one appliance does not guarantee all of them have been removed.

17. Lessons for AWS, Azure, Google Cloud, and Other Platforms

This incident does **not** demonstrate that CVE-2026-123890 affects AWS, Azure, or Google Cloud.

The vulnerability is specific to VMware vCenter.

What it does provide are architectural lessons for any environment where a control plane is separate from the resources it administers.

AWS

In AWS, administrative capabilities over EC2 resources depend on control-plane services and permissions, principally IAM and the relevant APIs.

Compromised credentials with sufficient permissions may enable sensitive operations on instances, volumes, and snapshots without using credentials within the guest operating system.

This does not suggest that AWS Nitro has VMware's vulnerability or that the two security models are equivalent.

The lesson is to place particularly strong protection around the identities, APIs, and permissions that govern infrastructure.

Microsoft Azure

In Azure, identities and roles with sufficient privileges over virtual machines, disks, and infrastructure resources may perform administrative operations from the control plane.

Defensive strategies should therefore include auditing operations, controlling privileges, and monitoring access to sensitive resources.

Google Cloud

In Google Cloud, identities with administrative permissions over Compute Engine and other services can modify resources through APIs.

IAM protection, audit logs, and restrictions on storage and administrative operations are fundamental.

Kubernetes

The parallel is also useful for Kubernetes.

Compromise of its control plane or highly privileged identities can allow an attacker to alter workloads and reach resources without individually compromising every container.

The Shared Lesson

The right question is not only:

“Is the operating system protected?”

We must also ask:

“Who can administer that system from the outside, and what would happen if that authority were compromised?”

18. Lessons Learned: Seven Changes This Incident Should Drive

Lesson 1. The Control Plane Is a Priority Target

Management infrastructure is not an auxiliary component. It may be one of the most critical parts of the entire architecture.

Lesson 2. An Internal Network Is Not Necessarily a Trusted Network

Initial access occurred through a product other than VMware. Lateral movement made it possible to reach services that may not have been exposed to the internet.

Lesson 3. Zero-Days Make Patch-Only Strategies Insufficient

Patching is mandatory, but it does not replace controls that impede exploitation and lateral movement.

Lesson 4. Identity Compromise Can Survive Password Changes

If signing keys are extracted or hidden administrative accounts created, recovery must address the affected trust mechanisms.

Lesson 5. Guest EDR Does Not Necessarily See Hypervisor Activity

Detection coverage must extend to APIs, management services, hypervisors, and storage.

Lesson 6. Infrastructure Logs Are First-Class Evidence

Without vCenter, ESXi, SSO, and network logs, a DFIR team may be unable to reconstruct how information was accessed.

Lesson 7. TTPs Matter More Than Ephemeral Indicators

Filenames, hashes, and IP addresses change. The behaviors required to compromise and abuse a control plane are harder to replace entirely.

19. Recommended Action Plan for a Security Team

Immediate Priority

1. Inventory vCenter instances and verify running versions.
2. Apply official fixes for CVE-2026-123890.
3. Review BeyondTrust Remote Support exposure and CVE-2026-102764 remediation status.
4. Check which systems can reach TCP/2012 and other management interfaces.

5. Hunt for unauthorized SSO accounts and SSH keys.
6. Review indicators and artifacts published by GTIG.

Short-Term Priority

1. Centralize and protect VCSA, ESXi, and SSO logs.
2. Set alerts for SSH enablement and creation of privileged accounts.
3. Separate administrative workstations from the broader network.
4. Apply restrictive access policies to the control plane.
5. Validate the SOC's practical ability to investigate virtual-disk operations.

Structural Priority

1. Classify the virtualization control plane as critical infrastructure.
2. Incorporate vCenter compromise scenarios into incident-response exercises.
3. Test recovery of identities and infrastructure from trusted states.
4. Review permissions granted to support, backup, and orchestration tools.
5. Measure detection coverage by ATT&CK technique, not simply by the security products deployed.

20. Conclusion: Protecting the Operating System Is No Longer Enough

ARMORY CASTLE stands out not only because it exploited two zero-day vulnerabilities.

It stands out because of the way the actor leveraged initial access to advance toward increasingly strategic components of the infrastructure.

First, it compromised a remote-support solution.

Next, it accessed an internal Linux appliance.

It then exploited vCenter, obtained administrative capabilities, compromised identity mechanisms, and reached ESXi.

Finally, it used specialized tools to extract information from virtual disks.

The operation demonstrates how an advanced attacker can deliberately seek out areas where traditional defenses have less visibility.

The attacker does not necessarily need to install malware inside the server holding the data. They can attempt to access those data from the infrastructure governing that server.

The defensive answer is not to abandon EDR, MFA, or patching. All remain essential.

It is to recognize that they represent only part of the security architecture.

> **The real lesson is that protecting machines is not enough: we must also protect, monitor, and be able to investigate everything that has authority over them.**

That lesson will remain valid long after the vulnerabilities in this incident have been fixed.

Bibliography and Technical Sources

[1] Google Threat Intelligence Group and Mandiant. *ARMORY CASTLE Exploits vSphere*. November 19, 2026. Primary source for the technical analysis, tools, TTPs, and indicators.
https://cloud.google.com/blog/topics/threat-intelligence/armory-castle-exploits-vsphere

[2] Broadcom. *VMSA-2026-0009.1: VMware vCenter, Tools, and SDDC Manager updates address multiple vulnerabilities*. Published November 19, 2026; updated November 20. Official source for severity, affected products, and corrected versions.
https://support.broadcom.com/web/ecx/support-content-notification/-/external/content/SecurityAdvisories/0/39495

[3] Google Threat Intelligence Group and Mandiant. *ARMORY CASTLE Exploits BeyondTrust*. October 15, 2026. Investigation into initial access and exploitation of CVE-2026-102764.
https://cloud.google.com/blog/topics/threat-intelligence/armory-castle-exploits-beyondtrust

[4] Mandiant. *vSphere and BRICKSTORM Malware: A Defender's Guide*. April 2, 2026. Background on VMware defensive architecture, logging, and hardening.
https://cloud.google.com/blog/topics/threat-intelligence/vsphere-brickstorm-defender-guide

[5] Mandiant. *VCSA Hardening & Logging Tool*. Technical reference repository. Deployment requires testing and compatibility review.
https://github.com/mandiant/vcsa-hardening-tool

[6] Mandiant. *VMware ESXi Zero-Day Used by Chinese Espionage Actor to Perform Privileged Guest Operations on Compromised Hypervisors*. June 13, 2023. Relevant precedent for attacks targeting the virtualization layer.
https://cloud.google.com/blog/topics/threat-intelligence/vmware-esxi-zero-day-bypass

[7] Mandiant. *Chinese Espionage Group UNC3886 Found Exploiting CVE-2023-34048 Since Late 2021*. January 19, 2024. Historical context on VMware exploitation by cyberespionage actors.
https://cloud.google.com/blog/topics/threat-intelligence/chinese-vmware-exploitation-since-2021

[8] MITRE ATT&CK. Knowledge base of adversary tactics and techniques.
https://attack.mitre.org/

[9] David Bianco. *The Pyramid of Pain*. Conceptual framework for prioritizing detection based on the cost imposed on the attacker.
https://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html

Methodological Note

This report is based on public information available as of **November 20, 2026**.

Facts attributed to ARMORY CASTLE are primarily drawn from the investigations by GTIG and Mandiant. The geopolitical attribution assessment belongs to those researchers; it does not constitute a judicial identification or publicly conclusive proof of state sponsorship.

The MITRE ATT&CK mapping, threat-hunting hypotheses, Sigma rules, and prioritization recommendations are defensive analysis developed from the published evidence.

Sigma rules require validation, field mapping, and testing in the target environment before deployment as production detections.