Kalinga University

Course- BCSAIMLCS/BCAAIML **Sem-IV**
Subject- Cloud Computing and Its Security
Subject Code: BCSAIMLCS403 & BCAAIML402

Unit IV

Isolation of users/VMs from each other/ what is Virtual Machine Based Isolation?

Virtual Machine-Based Isolation refers to a security and operational architecture where each virtual machine (VM) is isolated from others, ensuring that the activities, processes, and data within one VM cannot affect or compromise another VM or the host system. This concept is widely used in cloud computing, virtualization, and multi-tenant systems to protect resources and enhance security.

Key Aspects of VM-Based Isolation:

1. **Hypervisor Enforced Boundaries**:
   - A hypervisor (Type 1 or Type 2) creates and manages VMs and ensures they operate independently.
   - Each VM runs its own operating system and applications within a virtualized environment.
2. **Memory Isolation**:
   - VMs cannot directly access the physical memory or virtual memory of another VM.
   - Memory regions are allocated and controlled by the hypervisor.
3. **CPU and Execution Isolation**:
   - Each VM operates on virtual CPUs, ensuring no direct interference in CPU time or processing tasks between VMs.
4. **Storage Isolation**:
   - VMs typically use virtual disks, which are stored as files on the host system. These disks are isolated from each other to prevent unauthorized access or data leakage.
5. **Network Isolation**:
   - Virtual networks or VLANs are often used to segregate network traffic between VMs.
   - Firewalls and security policies are applied at the hypervisor or VM level.
6. **Crash or Fault Containment**:
   - A failure in one VM (e.g., due to a crash or malicious activity) does not propagate to others or the host system, providing fault tolerance.

Benefits of VM-Based Isolation:

- **Security**: Malicious activity in one VM is contained and cannot directly impact other VMs or the host system.
- **Multi-Tenancy**: Enables safe operation of workloads from multiple tenants on the same physical hardware in a shared environment.
- **Fault Tolerance**: Isolates software bugs or crashes within a single VM, preventing system-wide issues.

- **Flexibility**: Different operating systems and application stacks can coexist on the same physical hardware.

Applications of VM-Based Isolation:

- **Cloud Computing**: Public clouds use VM isolation to securely host workloads for different customers on shared hardware.
- **Testing and Development**: Developers run multiple isolated environments for testing without affecting other systems.
- **Secure Work Environments**: Virtual desktops ensure corporate and personal data stay isolated.

Virtual Machine-Based Isolation vs. Container-Based Isolation:

While VMs rely on hardware-level abstraction and a hypervisor, **containers** isolate workloads at the operating system level. Containers are more lightweight but less isolated compared to VMs. VM-based isolation is preferred for workloads requiring stringent security and multi-tenant environments.

Virtual Machine Security in Cloud

The term **"Virtualized Security,"** sometimes known as "security virtualization," describes security solutions that are software-based and created to operate in a virtualized IT environment. This is distinct from conventional hardware-based network security, which is static and is supported by equipment like conventional switches, routers, and firewalls.
Virtualized security is flexible and adaptive, in contrast to hardware-based security. It can be deployed anywhere on the network and is frequently cloud-based so it is not bound to a specific device.
In Cloud Computing**,** where operators construct workloads and applications on-demand, virtualized security enables security services and functions to move around with those on-demand-created workloads. This is crucial for virtual machine security. It's crucial to protect virtualized security in cloud computing technologies such as isolating multitenant setups in public cloud settings. Because data and workloads move around a complex ecosystem including several providers, virtualized security's flexibility is useful for securing hybrid and multi-cloud settings.

Service Provider Security

The system's virtualization hardware shouldn't be physically accessible to anyone not authorized. Each VM can be given an access control that can only be established through the Hypervisor in order to safeguard it against unwanted access by Cloud administrators. The three

fundamental tenets of access control, identity, authentication, and authorization**,** will prevent unauthorized data and system components from being accessed by administrators.

Hypervisor Security

The Hypervisor's code integrity is protected via a technology called Hyper safe. Securing the write-protected memory pages, expands the hypervisor implementation and prohibits coding changes. By restricting access to its code, it defends the Hypervisor from control-flow hijacking threats**.** The only way to carry out a VM Escape assault is through a local physical setting. Therefore, insider assaults must be prevented in the physical Cloud environment. Additionally, the host OS and the interaction between the guest machines need to be configured properly.

Virtual Machine Security

The administrator must set up a program or application that prevents virtual machines from consuming additional resources without permission. Additionally, a lightweight process that gathers logs from the VMs and monitors them in real-time to repair any **VM tampering must operate on a Virtual Machine**. Best security procedures must be used to harden the guest OS and any running applications. These procedures include setting up firewalls, host intrusion prevention systems (HIPS), anti-virus and anti-spyware programmers, online application protection, and log monitoring in guest operating systems.

Guest Image Security

A policy to control the creation, use, storage, and deletion of images must be in place for organizations that use virtualization. To find viruses, worms, spyware, and rootkits that hide from security software running in a guest OS, image files must be analyzed.

Benefits of Virtualized Security

Virtualized security is now practically required to meet the intricate security requirements of a virtualized network, and it is also more adaptable and effective than traditional physical security.

- **Cost-Effectiveness:** Cloud computing's virtual machine security enables businesses to keep their networks secure without having to significantly raise their expenditures on pricey proprietary hardware. Usage-based pricing for cloud-based virtualized security services can result in significant savings for businesses that manage their resources effectively.
- **Flexibility:** It is essential in a virtualized environment that security operations can follow workloads wherever they go. A company is able to profit fully from virtualization while simultaneously maintaining data security thanks to the protection it offers across various data centers, in multi-cloud, and hybrid-cloud environments.
- **Operational Efficiency:** Virtualized security can be deployed more quickly and easily than hardware-based security because it doesn't require IT, teams, to set up and configure several hardware appliances. Instead, they may quickly scale security systems by setting them up using centralized software. Security-related duties can be automated when security technology is used, which frees up more time for IT employees.
- **Regulatory Compliance:** Virtual machine security in cloud computing is a requirement for enterprises that need to maintain regulatory compliance because traditional hardware-based security is static and unable to keep up with the demands of a virtualized network.

Virtualization Machine Security Challenges

- As we previously covered, buffer overflows are a common component of classical network attacks. **Trojan horses, worms, spyware, rootkits, and DoS attacks** are examples of malware.
- In a cloud context, more recent assaults might be caused via VM rootkits, hypervisor malware, or guest hopping and hijacking. Man-in-the-middle attacks against VM migrations are another form of attack**.** Typically, passwords or sensitive information are stolen during passive attacks. Active attacks could alter the kernel's data structures, seriously harming cloud servers.
- **HIDS or NIDS** are both types of IDSs. To supervise and check the execution of code, use programmed shepherding. The **RIO dynamic optimization infrastructure**, the v Safe and v Shield tools from VMware, security compliance for hypervisors, and Intel vPro technology are some further protective solutions.

General ESXi Security Recommendations

Built-In Security Features

Risks to the hosts are mitigated as follows:

- ESXi Shell and SSH interfaces are disabled by default. Keep these interfaces disabled unless you are performing troubleshooting or support activities. For day-to-day activities, use the vSphere Client, where activity is subject to role-based access control and modern access control methods.
- Only a limited number of firewall ports are open by default. You can explicitly open additional firewall ports that are associated with specific services.
- ESXi runs only services that are essential to managing its functions. The distribution is limited to the features required to run ESXi.
- By default, all ports that are not required for management access to the host are closed. Open ports if you need additional services.
- By default, weak ciphers are disabled and communications from clients are secured by SSL. The exact algorithms used for securing the channel depend on the SSL handshake. Default certificates created on ESXi use PKCS#1 SHA-256 with RSA encryption as the signature algorithm.
- An internal web service is used by ESXi to support access by Web clients. The service has been modified to run only functions that a Web client requires for administration and monitoring. As a result, ESXi is not vulnerable to web service security issues reported in broader use.
- VMware monitors all security alerts that can affect ESXi security and issues a security patch if needed. You can subscribe to the VMware Security Advisories and Security Alerts mailing list to receive security alerts. See the webpage at http://lists.vmware.com/mailman/listinfo/security-announce.
- Insecure services such as FTP and Telnet are not installed, and the ports for these services are closed by default.
- To protect hosts from loading drivers and applications that are not cryptographically signed, use UEFI Secure boot. Enabling Secure Boot is done at the system BIOS. No additional configuration changes are required on the ESXi host, for example, to disk partitions. See UEFI Secure Boot for ESXi Hosts.
- If your ESXi host has a TPM 2.0 chip, enable and configure the chip in the system BIOS. Working together with Secure Boot, TPM 2.0 provides enhanced security and trust assurance rooted in hardware. See Securing ESXi Hosts with Trusted Platform Module.

Data storage considerations

CDP, which includes Time Travel and Fail-safe, is a standard set of features available to all Snowflake accounts at no additional cost. However, because your account is charged for all data stored in tables, schemas, and databases created in the account, CDP does have an impact on storage costs, based on the total amount of data stored and the length of time the data is stored.

Storage is calculated and charged for data regardless of whether it is in the Active, Time Travel, or Fail-safe state. Because these life-cycle states are sequential, updated/deleted data protected by CDP will continue to incur storage costs until the data leaves the Fail-safe state.

Monitoring data storage

Storage for your account (account administrators only)

If you have been assigned the ACCOUNTADMIN role (i.e. you serve as the top-level administrator for your Snowflake account), you can use Snowsight or the Classic Console to view data storage across your entire account:

Snowsight

Select **Admin** » **Cost Management** » **Consumption**.

Classic Console

Click on **Account** » **Billing & Usage** » **Average Storage Used**

This page displays the total average data storage for your account, as well as the total for all databases, internal and named stages, and data in Fail-safe.

For more information, see Exploring storage cost.

Individual table storage

Any user with the appropriate privileges can view data storage for individual tables. Snowflake provides the following methods for viewing table data storage:

Classic Console

Click on **Databases** » *<db_name>* » **Tables**

SQL

Execute a SHOW TABLES command.

or

Query either of the following:

- TABLE_STORAGE_METRICS view (in the Snowflake Information Schema).
- TABLE_STORAGE_METRICS view view (in Account Usage).

Of the three methods, TABLE_STORAGE_METRICS provides the most detailed information because it includes a breakdown of the physical storage (in bytes) for table data in the following three states of the CDP life-cycle:

- Active (ACTIVE_BYTES column)
- Time Travel (TIME_TRAVEL_BYTES column)
- Fail-safe (FAILSAFE_BYTES column)

The view also provides columns for distinguishing between owned storage and referenced storage that occurs when cloning tables (see section below).

Staged file storage (for data loading)

To support bulk loading of data into tables, Snowflake utilizes stages where the files containing the data to be loaded are stored. Snowflake supports both internal stages and external stages.

Data files staged in Snowflake internal stages are not subject to the additional costs associated with Time Travel and Fail-safe, but they do incur standard data storage costs. As such, to help manage your storage costs, Snowflake recommends that you monitor these files and remove them from the stages once the data has been loaded and the files are no longer needed. You can choose to remove these files either during data loading (using the COPY INTO <table> command) or afterwards (using the REMOVE command).

For more information, see Data loading considerations.

Tip
Periodic purging of staged files can have other benefits, such as improved data loading performance.

Cloned table, schema, and database storage

Snowflake's zero-copy cloning feature provides a convenient way to quickly take a "snapshot" of any table, schema, or database and create a derived copy of that object which initially shares the underlying storage. This can be extremely useful for creating instant backups that do not incur any additional costs (until changes are made to the cloned object).

However, cloning makes calculating total storage usage more complex because each clone has its own separate life-cycle. This means that changes can be made to the original object or the clone independently of each other and these changes are protected through CDP.

For example, when a clone is created of a table, the clone utilizes no data storage because it shares all the existing micro-partitions of the original table at the time it was cloned; however, rows can then be added, deleted, or updated in the clone independently from the original table. Each change to the clone results in new micro-partitions that are owned exclusively by the clone and are protected through CDP.

In addition, clones can be cloned, with no limitations on the number or iterations of clones that can be created (e.g. you can create a clone of a clone of a clone, and so on), which results in a n-level hierarchy of cloned objects, each with their own portion of shared and independent data storage.

Table IDs

Every Snowflake table has an ID that uniquely identifies the table. In addition, every table is also associated with a CLONE_GROUP_ID. If a table has no clones, then the ID and CLONE_GROUP_ID are identical. These IDs are displayed in the TABLE_STORAGE_METRICS view.

Owned storage versus referenced storage

When a table is cloned, it is assigned a new ID and the CLONE_GROUP_ID for the original table. At the instant the clone is created, all micro-partitions in both tables are fully shared. The storage associated with these micro-partitions is ***owned*** by the oldest table in the clone group and the clone ***references*** these micro-partitions.

After a clone is created, both tables within the clone group have separate life-cycles, such that any DML operations on either table create new micro-partitions that are owned by their respective tables. Storage associated with these micro-partitions can be queried using the RETAINED_FOR_CLONE_BYTES column in the TABLE_STORAGE_METRICS view.

Because every table within a clone group has an independent life-cycle, ownership of the storage within these tables sometimes needs to be transferred to a different table within the clone group. For example, consider a clone group that consists of:

Original table:		Cloned to:		Cloned to:
T1	»	**T2**	»	**T3**

If T2 and T3 share some micro-partitions and T2 is dropped, then ownership of that storage must be transferred before T2 enters Fail-safe. In Snowflake, this transfer occurs at the time the micro-partitions exit the Time Travel state and would otherwise enter Fail-safe. In the case above, the micro-partitions that were previously owned by T2 are transferred to T3 when the Time Travel retention period expires.

Managing costs for short-lived tables

CDP is designed to provide long-term protection for your data. This data is typically stored in permanent tables. Unless otherwise specified at the time of their creation, tables in Snowflake are created as permanent.

During an ETL or data modeling process, tables may be created that are short-lived. For these tables, it does not make sense to incur the storage costs of CDP. Snowflake provides two separate mechanisms to support short-lived tables:

- Temporary tables
- Transient tables

Temporary tables

Similar to other SQL databases, a temporary table exists only within a single user session and only within the duration of the session. Snowflake temporary tables have no Fail-safe and have a Time Travel retention period of only 0 or 1 day; however, the Time Travel period ends when the table is dropped.

Thus, the maximum total CDP charges incurred for a temporary table are 1 day (or less if the table is explicitly dropped or dropped as a result of terminating the session). During this period, Time Travel can be performed on the table.

Transient tables

Transient tables are unique to Snowflake. They have characteristics of both permanent and temporary tables:

- In contrast to temporary tables, transient tables are not associated with a session; they are visible to all users who have permissions to access that table. Also, similar to permanent tables, they persist beyond the session in which they were created.
- In keeping with temporary tables, transient tables have no Fail-safe and have a Time Travel retention period of only 0 or 1 day.

Thus, the maximum total CDP charges incurred for a transient table are 1 day. During this period, Time Travel can be performed on the table.

How partitioning models influence backup and recovery

Multi-tenant solutions can complicate database backup and recovery. AWS offers generic tools to help with backup and recovery, but you will probably have to leverage the tools provided by your database vendor to achieve the best results.

Silo-based SaaS deployment models do not have unique database backup and recovery restrictions beyond non-SaaS workloads.

Pooled and Hybrid SaaS deployment models bring challenges and we can generically approach them either by segregating (partitioning) the tenant data during the backup (extract) phase, or we can segregate the tenant data during the recovery (load) phase.

The operational effort and complexity of performing this segregation varies depending on the database partitioning model. The greater the resource sharing, the more effort required to partition a single tenant.

SaaS providers often choose the pooled model because it can be more cost-effective and simpler to operate for many tenants, however tenant backup and recovery is significantly more complex because of other tenants in the database compared to the silo model, where each tenant has their own database.

The partitioning model also influences the tools you can use for segregation. AWS service features, such as those from Amazon RDS, Aurora, or AWS Backup, back up the entire database instance (Amazon RDS) or cluster (Aurora). Where tenants are sharing resources, such as in the bridge or pool models, you must use database-engine specific tools like pg_dump or mysqldump to segregate tenant data, introducing additional tooling into your application.

We prefer segregation during recovery to take advantage of AWS service features during backup, and only introducing operational overhead during the recovery process. However, there are use cases for segregation during backup. Let's compare both approaches in more detail.

Segregating tenant data during backup (extract)

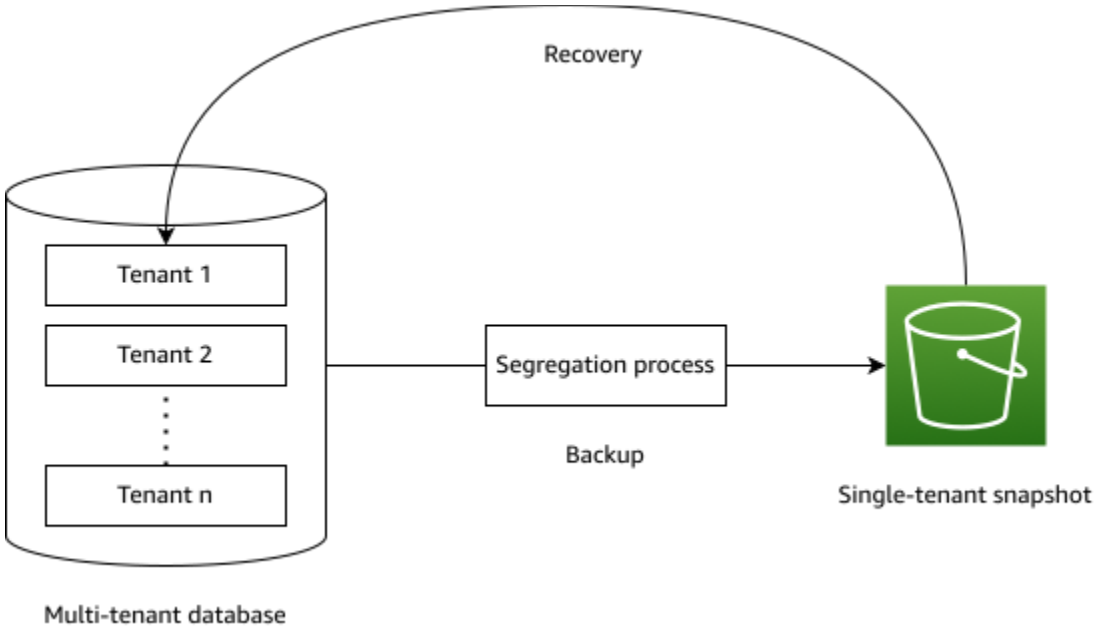
For SaaS builders, this approach offers simpler and quicker recovery, as well as per-tenant data retention. This incurs an operational overhead per-tenant cost during the backup cycle, which may make this approach unsuitable for SaaS providers with a large number of tenants.

Per-tenant data retention provides several product opportunities as a SaaS provider. You can offer customers the ability to manage their own retention policies, export historical data on demand, and encrypt their backups with their own encryption keys. Per-tenant data retention can also help improve your security and regulatory compliance postures.

From an operational perspective, per-tenant data retention provides simple mechanisms for managing a tenant's data. You can restore directly from backup, and a SaaS provider can easily delete data if a tenant leaves or invokes a "right to be forgotten" law.

Per-tenant data retention also provides an easy way to measure per-tenant backup costs.

To segregate during backup, each tenant has their data segregated in the backup process and stored independently, as shown in the following diagram. The methods available to segregate a tenant's data depend on the partitioning model, which we investigate later.



Segregating tenant data during backup adds operational overhead to each backup cycle. This may not be practical for SaaS providers with many tenants. You must manage each tenant backup over time, and the work needed to perform segregation brings additional performance and financial cost to each backup cycle.

To ensure data consistency in your backups, consult the best practices of your chosen relational database. This may require read only locks, using write ahead buffers, or service downtime. These choices impact your tenants' experience and add further complexity to the backup process.

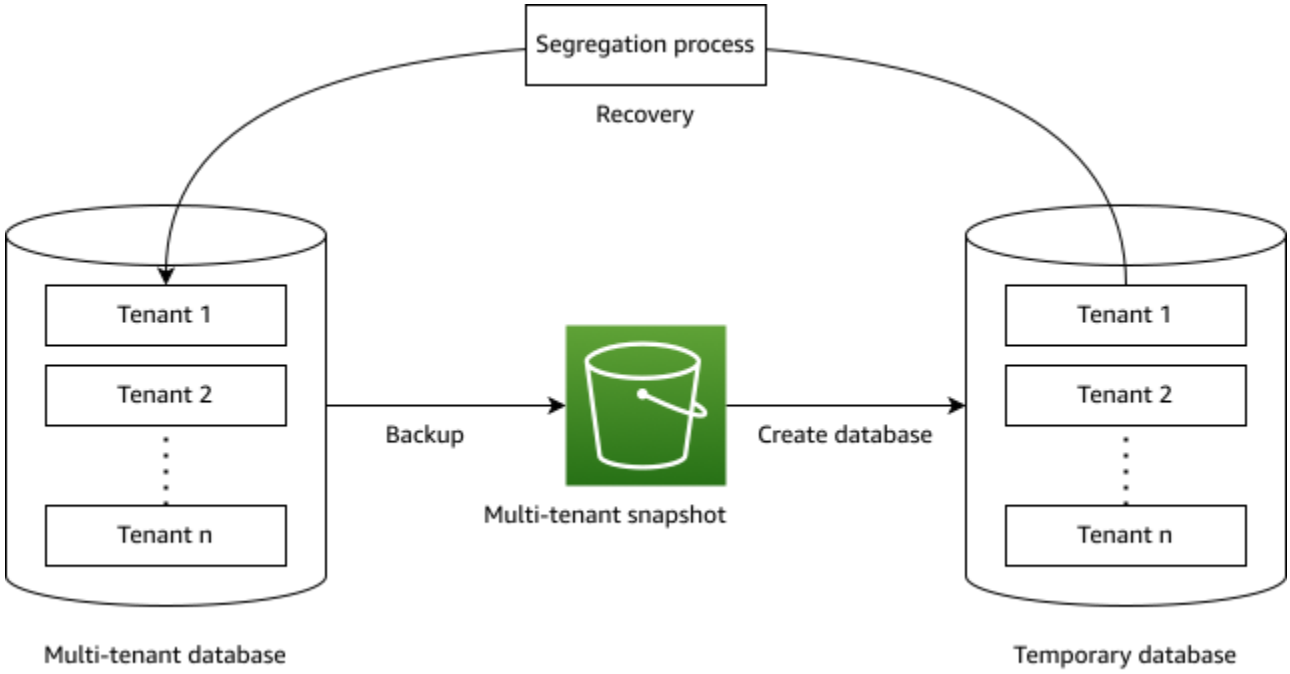
Segregating tenant data during recovery (load)

Segregation during recovery is attractive to SaaS builders because it is operationally efficient and you can incorporate it into an existing backup process.

This approach has the benefit that you can use AWS service features for backup. Aurora backups are continuous and incremental in nature, without causing a performance impact to your application or any interruption to the database service. This is important for a SaaS offering, which should be always available for tenants.

You introduce complexity from segregation only in the recovery process and do not require service downtime during backup. However, this can add performance and usability issues for your tenants during recovery.

In segregation during recovery, you backup the entire database containing all tenants. You restore the entire multi-tenant snapshot to a temporary database, where you then segregate the tenant and restore their data into the live database. You can then delete the temporary database. The following diagram illustrates this process.



The cost of the temporary database needs to be considered during the recovery process. You must take care to reduce costs, especially for large databases. We investigate these options later in this post.

Restoring tenant data to a live database

During the recovery process, you must think about how to handle existing tenant data and how to restore recovered data back into the multi-tenant database.

A complete restoration to an earlier point in time is the simplest method. Delete all existing tenant data before restoration. Complexity of recovery in this scenario depends on the partitioning model, which we explore later.

For selective recovery, you need to overwrite any existing data that is being restored. You can use database engine-specific tools to import the segregated tenant data into the live database, overwriting existing files.

Now that we've described some benefits and challenges to the two main approaches to backup and restore of multi-tenant data, let's look at some examples of how you might implement them with PostgreSQL.

Critical Virtualization Vulnerabilities

Some attacks against virtual machine, or VM, environments are variations of common threats such as denial of service. Others are still largely theoretical but likely approaching as buzz and means increase. Keep an eye on these critical weaknesses:

VM sprawl:VMs are easy to deploy, and many organizations view them as hardware-like tools that don't merit formal policies.This has led to VM sprawl, which is the unplanned proliferation of VMs.Attackers can take advantage of poorly monitored resources.More deployments also mean more failure points, so sprawl can cause problems even if no malice is involved.

Hyperjacking: Hyperjacking takes control of the hypervisor to gain access to the VMs and their data. It is typically launched against type 2 hypervisors that run over a host OS although type 1 attacks are theoretically possible. In reality, hyperjackings are rare due to the difficulty of directly accessing hypervisors.However, hyperjacking is considered a real-world threat, and administrators should take the offensive and plan for it.

VM escape:A guest OS escapes from its VM encapsulation to interact directly with the hypervisor.This gives the attacker access to all VMs and, if guest privileges are high enough, the host machine as well. Although few if any instances are known, experts consider VM escape to be the most serious threat to VM security.

Denial of service: These attacks exploit many hypervisor platforms and range from flooding a network with traffic to sophisticated leveraging of a host's own resources.The availability of botnets continues to make it easier for attackers to carry out campaigns against specific servers and applications with the goal of derailing the target's online services.

Incorrect VM isolation: To remain secure and correctly share resources,VMs must be isolated from each other.Poor control over VM deployments can lead to isolation breaches in which VMs communicate.Attackers can exploit this virtual drawbridge to gain access to multiple guests and possibly the host.

Unsecured VM migration: This occurs when a VM is migrated to a new host, and security policies and configuration are not updated to reflect the change.Potentially, the host and other guests could become more vulnerable.Attackers have an advantage in that administrators are likely unaware of having introduced weaknesses and will not be on alert.

Host and guest vulnerabilities: Host and guest interactions can magnify system vulnerabilities at several points.Their operating systems, particularly Windows, are likely to have multiple weaknesses.Like other systems, they are subject to vulnerabilities in email, Web browsing, and

network protocols.However, virtual linkages and the co-hosting of different data sets make a serious attack on a virtual environment particularly damaging.

How to Mitigate Risk

Fortunately, security engineers can take several steps to minimize risk.The first task is to accurately characterize all deployed virtualization and any active security measures beyond built-in hypervisor controls on VMs.Security controls should be compared against industry standards to determine gaps.Coverage should include anti-virus, intrusion detection, and active vulnerability scanning.Additionally, consider these action steps:

VM traffic monitoring: The ability to monitor VM backbone network traffic is critical.Conventional methods will not detect VM traffic because it is controlled by internal soft switches.However, hypervisors have effective monitoring tools that should be enabled and tested.

Administrative control: Secure access can become compromised due to VM sprawl and other issues.Ensure that authentication procedures, identity management, and logging are ironclad.

Customer security: Outside of the VM, make sure protection is in place for customer-facing interfaces such as websites.

VM segregation: In addition to normal isolation, strengthen VM security through functional segregation.For example, consider creating separate security zones for desktops and servers.The goal is to minimize intersection points to the extent feasible.

Why do you need vulnerability management?

According to a Forrester report:

> "44% indicated that the most recent data breach involved more than one external attack method while 34% reported that their external breach involved software vulnerability exploits.

On top of that, over 38,500 vulnerabilities have been disclosed in 2024 so far, a steep 34% increase since 2023 - highlighting the importance for organizations to include vulnerability management and vulnerability remediation to improve their security posture.

Before implementing vulnerability remediation in your enterprise, it is imperative to understand the current landscape and the challenges faced in vulnerability management.

What are the challenges in vulnerability management?

In most organizations, there are literally too many vulnerabilities to track manually, and not all of them pose equal risk. Now, imagine tracking multiple vulnerabilities across thousands of heterogeneous assets in a distributed network. With the window between disclosure of vulnerabilities and their exploit by malicious actors shrinking, organizations need to be swift in their remediation.

With limited time and resources and without the risk background necessary to prioritize issues, your vulnerability management efforts may be futile. Adding to this, many vulnerability management tools in the market offer patching through a third-party integration, but juggling multiple tools for vulnerability assessment and patch management results in a fragmented and inefficient workflow.

If a malicious actor does use a vulnerability as a gateway into the network, it's the overlooked misconfigurations that they'll leverage to laterally move and exploit other machines within the network. This is why every loophole must be addressed along with software vulnerabilities to gain a strong security strategy and minimize the attack surface.

While issuing vendor-published patches to affected machines is the ideal remediation option, having a fail-safe plan to retreat to in case of unpatchable circumstances such as end-of-life software and zero-day vulnerabilities is essential.

Benefits of vulnerability management

With the exponential rise in the number of vulnerabilities in the last few years, the cyber forefront of organizations is at extreme risk. All that the threat actors need is one vulnerability exploitation, which is enough to land your enterprise into a tangle of cyber compliance fines and lost business.

If you're unsure how vulnerability management and timely vulnerability remediation can benefit your network, read below:

- **Improved visibility and control** over the managed endpoints, ensuring hardened security against exploits. Regular scans and patching schedules by vulnerability management tools can help identify vulnerabilities and weaknesses in the network.
- **Efficient cyber-risk management**, thereby negating human errors such as unrequired open ports, weak passwords, and so on.
- **Streamlined operations** by reducing downtime, data protection, and lesser recovery time, in the case of an exploit.
- **Adhering to regulatory compliances** with continual scanning, regular patching, in-depth reports, and enhanced audits.

What are the 4-steps of vulnerability management?

Vulnerability management, being a continual process works best when automated using a vulnerability management tool. Explained below are the 4-steps of vulnerability management.

Step 1: Asset Discovery

The asset scanning module of the vulnerability management software scans the network for newly added endpoints. On an endpoint level, it scans them to add or modify any newly added software or applications.

Step 2: Vulnerability Scanning

The network endpoints and installed applications are periodically scanned for vulnerable software, open ports, web server misconfigurations, and other vulnerabilities that can exploit the network.
The vulnerability scanner, which forms an integral part of any vulnerability management tool tallies a publicly available database of vulnerabilities and exploits to associate the known vulnerabilities with the ones scanned in the network endpoints and applications.

Step 3: Vulnerability Assessment

The next step in vulnerability management is prioritization. Prioritizing the vulnerabilities enables IT security teams to mitigate the ones that are most likely to cause an exploit in the network. One prioritization method is via CVSS scores (as mentioned in the table below).

CVSS v3.x Ratings		**CVSS v4.0 Ratings**	
Severity	**Severity Score Range**	**Severity**	**Severity Score Range**
Low	0.1-3.9	Low	0.1-3.9
Medium	4.0-6.9	Medium	4.0-6.9
High	7.0-8.9	High	7.0-8.9
Critical	9.0-10.0	Critical	9.0-10.0

While these scores help to prioritize mitigation, the risk possessed by certain vulnerabilities also depends on other factors, hence risk-based prioritization.

Step 4: Vulnerability Remediation

Last but not least comes mitigation. Once the vulnerabilities are detected and prioritized, they are mitigated by one of the following means:

- Deploying the latest patches to prevent further exploitation.
- Quarantining the vulnerable systems or uninstalling the vulnerable applications in case patches are not available or no fix has been released by the vendor.
- Marking as exclusions in case of known/internal vulnerabilities or when the chances of exploitation are less likely.

These four steps together make up vulnerability management. To bolster the enterprise network, it is recommended to automate the vulnerability management process.

4-step vulnerability management process

Each stage of the VM workflow plays a crucial role in reducing risk and enhancing overall security posture. By leveraging tools like vulnerability scanners, alinging with risk-based frameworks, and validating vulnerabilities, organizations can constantly refine their efforts to focus on addressing the threats that matter most while increasing operational resilience.

Step 1: Perform vulnerability scan

At the heart of a typical VM tool is a vulnerability scanner. The scan consists of four stages:

1. Scan network-accessible systems by pinging them or sending them TCP/UDP packets
2. Identify open ports and services running on scanned systems
3. If possible, remotely log in to systems to gather detailed system information
4. Correlate system information with known vulnerabilities

Vulnerability scanners are able to identify a variety of systems running on a network, such as laptops and desktops, virtual and physical servers, databases, firewalls, switches, printers, etc. Identified systems are probed for different attributes: operating system, open ports, installed software, user accounts, file system structure, system configurations, and more.

This information is then used to associate known vulnerabilities to scanned systems. In order to perform this association, vulnerability scanners will use a vulnerability and exploit database that contains a list of publicly known vulnerabilities.

Properly configuring vulnerability scans is an essential component of a VM solution. Vulnerability scanners can sometimes disrupt the networks and systems that they scan. If available network bandwidth becomes very limited during an organization's peak hours, then vulnerability scans should be scheduled to run during off hours.

If some systems on a network become unstable or behave erratically when scanned, they might need to be excluded from vulnerability scans, or the scans may need to be fine-tuned to be less disruptive.

Step 2: Vulnerability assessment

After vulnerabilities are identified, they need to be assessed so the risks posed by them are dealt with appropriately and in accordance with an organization's vulnerability management program framework. VM platforms will provide different risk ratings and scores for vulnerabilities, such as Common Vulnerability Scoring System (CVSS) scores.

These scores are helpful in telling organizations which vulnerabilities they should focus on first, but the true risk posed by any given vulnerability depends on some other factors beyond those out-of-the-box risk ratings and scores.

RBVM takes vulnerability assessment a step further by factoring in the criticality of the affected assets, the exploitability of the vulnerability, and the potential impact on the organization if it were exploited. It aligns VM efforts with an organization's unique risk tolerance and operational priorities, ensuring resources are allocated to address the most pressing threats first. By integrating RBVM principles, organizations can move beyond static scoring systems and develop a dynamic, context-aware strategy that keeps their most valuable assets secure.

Like any security tool, vulnerability scanners aren't perfect. Their vulnerability detection false-positive rates, while low, are still greater than zero. Performing vulnerability validation with penetration testing tools and techniques helps weed out false-positives so organizations can focus their attention on dealing with real vulnerabilities.

The results of vulnerability validation exercises or full-blown penetration tests can often be an eye-opening experience for organizations that thought they were secure enough or that the vulnerability wasn't that risky.

Step 3: Prioritize and remediate vulnerabilities

Once a vulnerability has been validated and deemed a risk, the next step is prioritizing how to treat that vulnerability:

- **Remediation**: Fully fixing or patching a vulnerability so it can't be exploited – this is ideal.
- **Mitigation:** Lessening the likelihood and/or impact of a vulnerability being exploited. This is sometimes necessary when a proper fix or patch isn't yet available for an identified vulnerability.
- **Acceptance:** Taking no action to fix or otherwise lessen the likelihood/impact of a vulnerability being exploited. This is typically justified when a vulnerability is deemed a low risk, and the cost of fixing the vulnerability is substantially greater than the cost incurred by an organization if the vulnerability were to be exploited.

When remediation activities are completed, it's best to run another vulnerability scan to confirm that the vulnerability has been fully resolved.

However, not all vulnerabilities need to be fixed. For example, if an organization's vulnerability scanner has identified vulnerabilities in Adobe Flash Player on their computers, but they completely disabled Adobe Flash Player from being used in web browsers and other client applications, then those vulnerabilities could be considered sufficiently mitigated by a compensating control.

Step 4: Continuous vulnerability management

Performing regular and continuous vulnerability assessments enables organizations to understand the speed and efficiency of their VM program over time. VM tools typically have different options for exporting and visualizing vulnerability scan data with a variety of customizable reports and dashboards.

Not only does this help IT teams easily understand which remediation techniques will help them fix the most vulnerabilities with the least amount of effort, or help security teams monitor vulnerability trends over time in different parts of their network, but it also helps support organizations' compliance and regulatory requirements**.**

Vulnerability management automation

A VM system can help automate the VM process, streamlining the identification, assessment, and treatment of vulnerabilities across an organization's attack surface. These systems typically use a combination of vulnerability scanners and endpoint agents to inventory systems on a network, identify vulnerabilities, and evaluate their potential impact. Let's take a look into how VM automation can play into different scenarios.

Advanced testing

To ensure vulnerabilities are addressed effectively, organizations can integrate automated VM with broader cybersecurity practices like penetration testing and breach and attack simulation (BAS).

Penetration testing validates vulnerabilities by simulating real-world attacks, providing critical insights into their exploitability and potential business impact.

Similarly, BAS tools help organizations continuously test their defenses, offering automated, scalable ways to identify weaknesses and evaluate the strength of existing controls.

Collaborative teams

VM automation can also benefit from incorporating the methodologies of red team, blue team, and purple team exercises. Automated tools can generate actionable data red teams use to simulate offensive tactics and assess vulnerabilities in systems.

This data can then inform blue teams in their efforts to bolster defenses and proactively mitigate risks. Purple teams, acting as a bridge, can leverage automated insights to facilitate collaboration between red and blue teams, ensuring both strategies are refined in real-time.

Exposure management

An automated VM program should not operate in isolation but as part of a broader exposure management strategy. By integrating with attack surface management (ASM) tools, organizations can gain continuous visibility into their dynamic digital footprints, including shadow IT and cloud environments.

The vulnerabilities of hypervisors

The efficiency of hypervisors against cyberattacks has earned them a reputation as a reliable and robust software application. But the persistence of hackers who never run out of creative ways to breach systems keeps IT experts on their toes. You should know the vulnerabilities of hypervisors so you can defend them properly and keep hackers at bay.

A hypervisor is a software application that distributes computing resources (e.g., processing power, RAM, storage) into virtual machines (VMs), which can then be delivered to other computers in a network. This gives people the resources they need to run resource-intensive applications without having to rely on powerful and expensive desktop computers.

System administrators can also use a hypervisor to monitor and manage VMs. So if hackers manage to compromise hypervisor software, they'll have unfettered access to every VM and the data stored on them.

While hypervisors are generally well-protected and robust, security experts say hackers will eventually find a bug in the software. So far, there have been limited reports of hypervisor hacks; but in theory, cybercriminals could run a program that can break out of a VM and interact directly with the hypervisor. From there, they can control everything, from access privileges to computing resources.

Another point of vulnerability is the network. Since hypervisors distribute VMs via the company network, they can be susceptible to remove intrusions and denial-of-service attacks if you don't have the right protections in place.

If those attack methods aren't possible, hackers can always break into server rooms and compromise the hypervisor directly. So what can you do to protect against these threats?

Create separate VM and management networks

Keeping your VM network away from your management network is a great way to secure your virtualized environment. If malware compromises your VMs, it won't be able to affect your hypervisor.

Set access privileges

Ideally, only you, your system administrator, or virtualization provider should have access to your hypervisor console. You need to set strict access restrictions on the software to prevent unauthorized users from messing with VM settings and viewing your most sensitive data.

Disable unnecessary services

Off-the-shelf operating systems will have many unnecessary services and apps that increase the attack surface of your VMs. If you can't tell which ones to disable, consult with a virtualization specialist.

Pay attention to physical security

Breaking into a server room is the easiest way to compromise hypervisors, so make sure your physical servers are behind locked doors and watched over by staff at all times.

Install top-notch network security tools

Due to network intrusions affecting hypervisor security, installing cutting-edge firewalls and intrusion prevention systems is highly recommended. These security tools monitor network traffic for abnormal behavior to protect you from the newest exploits.

Stay on top of hypervisor updates

Hypervisors must be updated to defend them against the latest threats. But if you'd rather spend your time on more important projects, you can always entrust the security of your hypervisors to a highly experienced and certified managed services provider, like us.

Contact us today to see how we can protect your virtualized environment.

Hypervisor escape vulnerabilities are security flaws that allow a virtual machine (VM) or malicious code within it to bypass the isolation provided by a hypervisor and access other VMs or the host system. These vulnerabilities pose significant risks in virtualized environments, as they undermine the core security model of hypervisors.

Key Aspects of Hypervisor Escape Vulnerabilities

1. **Nature of the Vulnerabilities**:
   - **Code Execution**: Exploiting weaknesses in the hypervisor code to execute arbitrary code at the hypervisor or host level.
   - **Isolation Breach**: Breaking the boundary that isolates VMs from each other and the host.
   - **Privilege Escalation**: Gaining administrative or root-level access to the host system.
2. **Potential Impact**:
   - **Data Breaches**: Accessing sensitive data from other VMs or the host.
   - **Denial of Service (DoS)**: Disrupting the operation of the host or other VMs.
   - **Lateral Movement**: Using the compromised hypervisor to attack other systems on the network.
3. **Common Causes**:
   - **Bugs in Hypervisor Code**: Errors in software implementations (e.g., VMware ESXi, Microsoft Hyper-V, Xen, or KVM).
   - **Misconfigurations**: Improper setup of virtual machines, hypervisor settings, or resource controls.
   - **Hardware Vulnerabilities**: Exploiting flaws in CPUs or memory management (e.g., speculative execution vulnerabilities like Meltdown and Spectre).
4. **Examples of Hypervisor Escape Vulnerabilities**:

- **Venom (CVE-2015-3456)**: A flaw in the virtual floppy disk controller used in many hypervisors allowed attackers to execute code on the host.
- **L1 Terminal Fault (L1TF)**: A hardware vulnerability affecting Intel CPUs that could be exploited to bypass isolation.
- **CVE-2021-26937**: A Microsoft Hyper-V vulnerability allowing remote code execution via crafted packet processing.

5. **Detection and Prevention**:
   - **Patching and Updates**: Regularly apply updates to the hypervisor and host OS to fix known vulnerabilities.
   - **Secure Configurations**: Limit VM access to critical resources and isolate untrusted VMs in separate networks or clusters.
   - **Monitoring and Intrusion Detection**: Use tools to monitor for unusual activity in VMs and the host.
   - **Hardened Hypervisor Settings**: Disable unnecessary features and use trusted boot mechanisms.
   - **Hardware-Assisted Virtualization Security**: Leverage features like Intel VT-d and AMD-V for enhanced security.
6. **Mitigation Strategies**:
   - Use hypervisors with strong track records of security and rapid patch response.
   - Employ least privilege principles, limiting user and process access to only what's necessary.
   - Implement network segmentation and firewalls to restrict communication between VMs.
   - Regularly audit configurations and logs for anomalies or misconfigurations.

When dealing with configuration issues related to malware, it's crucial to address the situation methodically to ensure the malware is properly removed and the system is restored securely. Here's a structured approach:

---

1. Identify Symptoms

- **Performance Issues**: Slow system performance, unexpected crashes.
- **Unauthorized Activities**: Unknown programs running, unexpected network activity.
- **Changes in Configuration**: Altered system settings or files.
- **Pop-ups or Warnings**: Frequent pop-ups, especially fake security alerts.

---

2. Isolate the System

- **Disconnect from the Network**: Prevent the malware from spreading or sending data.
- **Quarantine the Device**: Avoid connecting external drives or peripherals.

---

3. Scan and Remove Malware

- **Use Reputable Security Tools**: Tools like Malwarebytes, Windows Defender, or specialized anti-malware tools.
- **Run a Full Scan**: Perform a thorough scan to detect and remove malware.
- **Boot into Safe Mode** (if possible): This can prevent the malware from actively interfering with the removal process.

---

4. Check and Restore Configuration

- **Review Key Configurations**:
  - **System Startup Items**: Check for unauthorized programs (e.g., using Task Manager or msconfig on Windows).
  - **Registry Entries**: Look for malicious entries (be cautious—only edit the registry if you know what you're doing).
  - **Hosts File**: Ensure it hasn't been altered to redirect traffic.
  - **Browser Settings**: Remove unauthorized extensions or changes.
- **Reset System Policies**: Malware often alters group policy settings or permissions.
- **Restore Default Settings**: Use tools like "Reset this PC" on Windows or reinstall the OS if necessary.

---

5. Update Software

- **Install Updates**: Ensure the operating system, antivirus, and software are up to date.
- **Check Firmware**: In advanced cases, malware might infect firmware.

---

6. Strengthen Security

- **Install Endpoint Protection**: Use robust antivirus/antimalware software.
- **Enable Firewalls**: Both system and network-level firewalls.
- **Use Strong Passwords**: Update all system and account passwords.
- **Educate Users**: Train users to recognize phishing and malicious activities.

---

7. Monitor and Verify

- **Logs and Alerts**: Monitor system logs for unusual activity.
- **Third-Party Tools**: Use tools like Sysinternals Suite or ESET SysRescue for deeper verification.
- **Penetration Testing**: If possible, assess vulnerabilities to prevent recurrence.