Virtual Machine Generation ID with Active Directory Domain Controllers

ravenswoodtechnology.com/virtual-machine-generation-id-with-active-directory-domain-controllers

Josh Goben October 12, 2023

Active Directory (AD) domain controllers (DCs) have been around since Windows 2000. At that time, virtualization was in its infancy and almost every server was physical. And many of those servers weren't even housed in a typical datacenter or server closet.

Virtualization started to take off in the early to mid-2000s. It enabled the use of capabilities such as virtual machine (VM) snapshots and cloning, both of which presented several problems for DCs. Because AD wasn't built to support snapshots, restores, or cloning, performing those actions on a DC could break replication with the rest of the domain—or even cause bad data to replicate out.

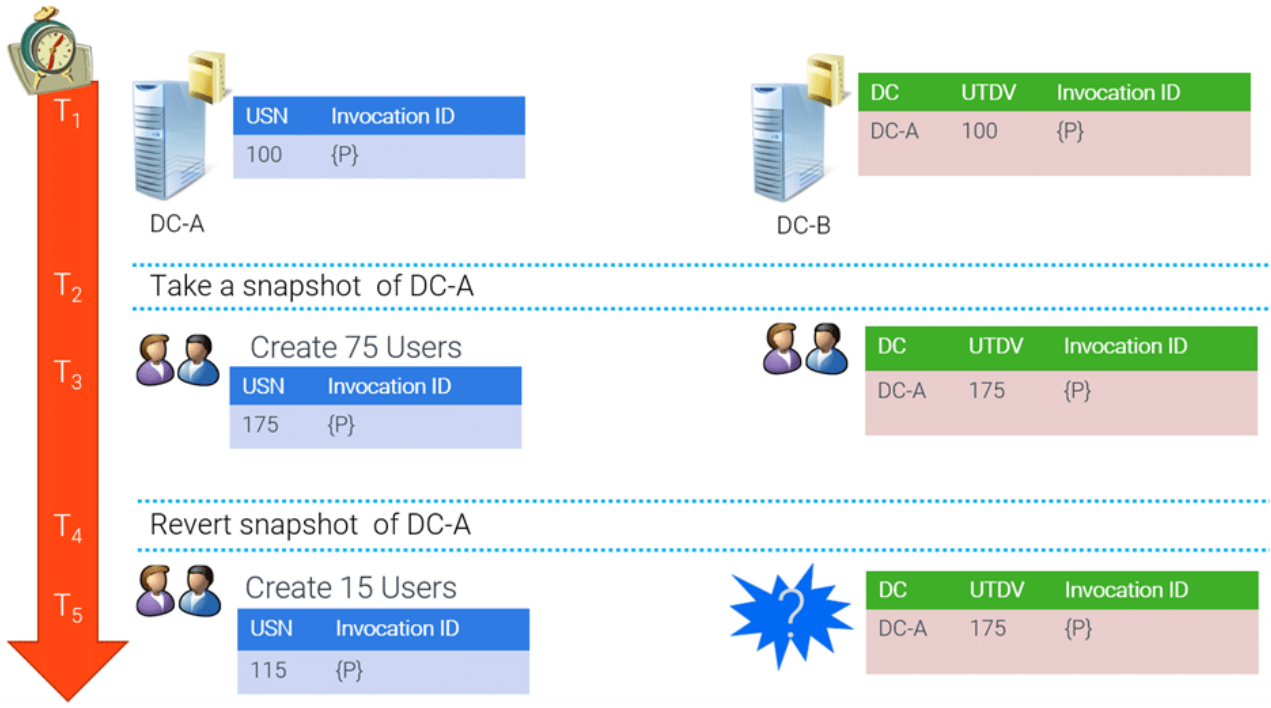
In response, Microsoft worked with hypervisor manufacturers to develop a concept called VM-Generation ID (VMGenID). VMGenID allows DCs to infer when they've been restored from a snapshot or clone.

USN Rollback

The update sequence number (USN) is managed on each DC to indicate how many updates have been made to its database. Every change (e.g., a password reset, a new user, an extensionAttribute modification) causes the USN to be incremented. This lets the DC keep track of how many updates it has made and lets other DCs know when there are new updates that need to be replicated. Like a clock, the USN can only ever move forward.

When a snapshot is applied on a virtualized DC, this allows you to roll back the clock. This is called USN rollback. When the snapshot is applied, any changes made since the snapshot was taken are lost—at least according to that DC. If the changes replicated out, then those changes won't automatically replicate back in, and you'll end up with replication inconsistency.

See the following illustration for an example of USN rollback impact on DC replication.



In this example, we see two DCs (DC-A and DC-B) where DC-B is replicating inbound from DC-A.

1. We start at USN 100; DC-B knows its last replicated changes from DC-A stopped at USN 100. DC-B keeps track of this in the up-to-dateness vector (UTDV).
2. We then take a snapshot of DC-A.
3. We create 75 new users on DC-A. This increases the highest committed USN on DC-A to 175, and DC-B replicates the users. DC-B updates its UTDV entry for DC-A to 175.
4. We then revert DC-A back to its snapshot, and the highest committed USN on DC-A reverts to 100.
5. We create 15 more users on DC-A, increasing DC-A's highest committed USN to 115. DC-B's UTDV says that the last change from DC-A was USN 175. Because of this, DC-B doesn't replicate the new users from DC-A since DC-B's UTDV says it has already received those changes.

At that point, replication from DC-A to DC-B won't occur until enough changes have been made that the highest committed USN on DC-A passes 175. And even then, only future changes replicate. This means the 15 new users we just created are stuck on DC-A, and the 75 users we initially created and replicated to DC-B won't replicate back to DC-A. This condition is called USN rollback.

SID Duplication

A security identifier (SID) is a unique value made up of two major parts: the domain SID and a relative ID (RID). The RID portion is a unique number assigned by a DC when a security principal is created in the domain. Each DC will have a pool of RID values available to assign. The RID pool is issued to DCs by the domain's RID master. This design ensures that no DC will ever have an overlapping pool of RIDs.

When a DC is rolled back in time, the rollback includes any RIDs that were issued after the snapshot was taken. If a new principal is created on that DC, it may reuse a RID that was already issued to another principal. At that point you will have multiple principals in the domain with the same SID, which is a security concern since the wrong principal could potentially gain access to unauthorized resources. AD protects itself by deleting all the objects with the duplicate SID.

VM-Generation ID

Starting with Windows Server 2012, DCs look at an attribute provided by the hypervisor, called VMGenID. Microsoft's Hyper-V and VMware were two of the first hypervisors to support VMGenID, but practically all enterprise hypervisors and cloud services now support VMGenID.

VMGenID is a unique value presented to the VM guest OS by the hypervisor itself. It's set when the VM is created and only changes when something big happens, such as a snapshot restore, a clone, or a failover event. It also resets every time you stop and deallocate a VM in Azure. You can read more about VMGenID in the Microsoft Hyper-V documentation for Server 2012.

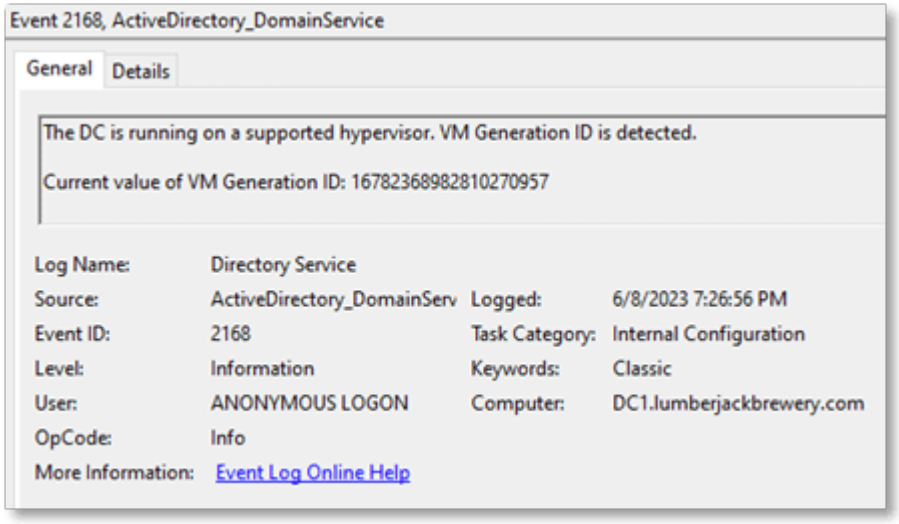
DCs keep track of their VMGenID value and will check it every time the system boots and prior to writing changes to the AD database. If the VMGenID value has changed, the DC infers that some kind of virtualization event has occurred. It will then automatically protect itself (more on this later).

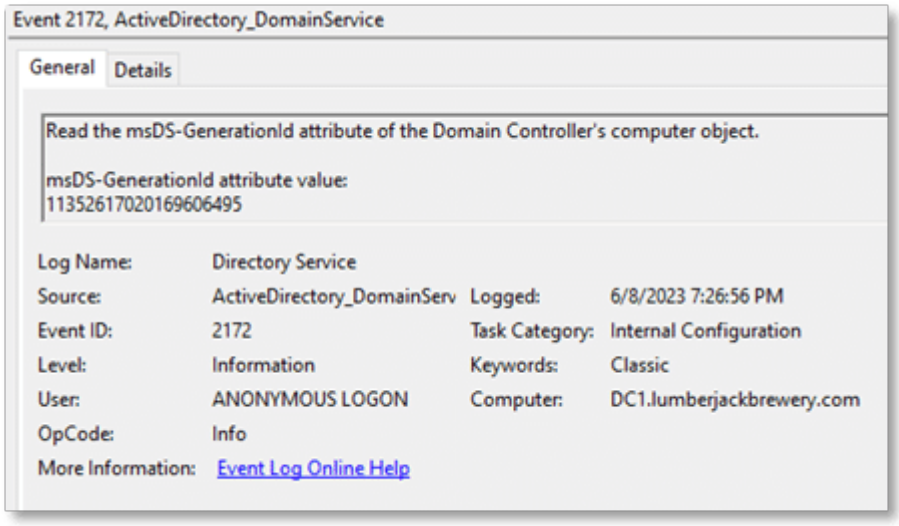
VMGenID Events

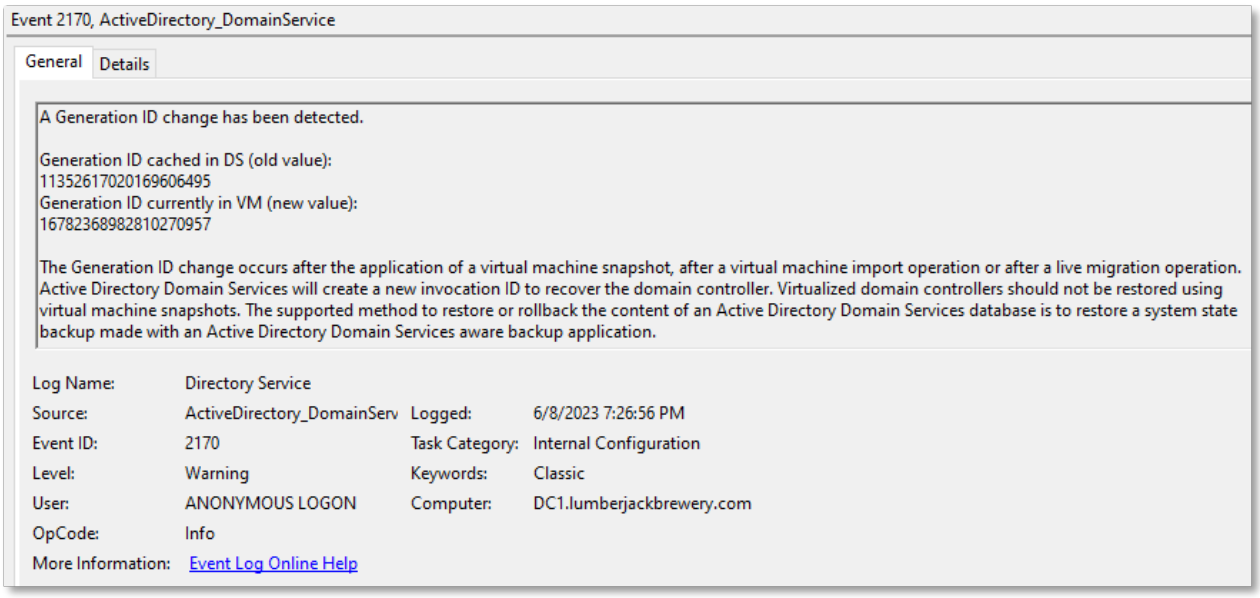
Several events are logged regarding VMGenID. These events are found in the Directory Services event log:

- **2168** – Indicates that VMGenID has been detected and read from the hypervisor
- **2172** – Indicates that previous VMGenID has been read from the DC's computer object in the local AD database
- **2170** – Indicates that a change in VMGenID has been detected and that the invocation ID is being reset

You should expect to see event 2168 and 2172 on every VM boot. Event 2170 isn't expected unless you performed a restore or cloning procedure, or unless you stopped and deallocated a VM in Azure.







Automatic DC Protection when VMGenID Changes

When a VMGenID change is detected by a DC, two main actions occur:

1. The DC's invocation ID is reset.

2. The current RID pool is released, and a new RID pool is requested from the RID master.

Invocation ID

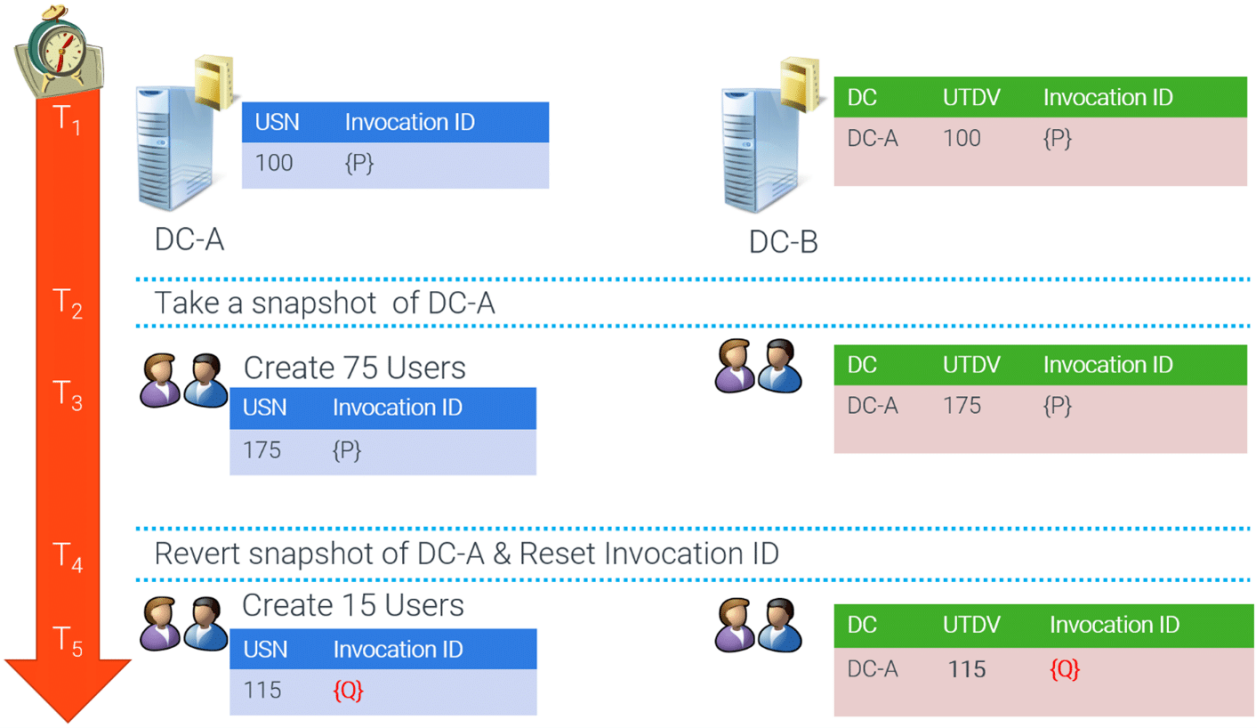
The invocation ID is a GUID that identifies an instance of the AD database. The invocation ID is how replication partners track partner DCs in their UTDV. When the invocation ID changes, a new entry is created in the UTDV. Resetting the invocation ID allows the snapshotted DC to go back in time but then start a new instance of the AD database and start inbound replication again.

RID Pool

When the RID pool is released, this action ensures that any RIDs issued after the initial snapshot won't be used again. Since a RID is a single-use identifier and the RID master only issues a RID set once (500 at a time, by default), then we can be assured that no RID re-use will occur—which means SID duplication has been avoided.

Catastrophe Averted

Let's go back and look at our previous example again, now that we have DCs that are aware of when a snapshot or cloning event has occurred.



In this new example, DC-A and DC-B are running at least Server 2012.

1. We start again with a highest committed USN of 100 on DC-A.
2. We take a snapshot of DC-A.
3. We create 75 new users on DC-A, increasing the highest committed USN to 175. DC-B replicates the new users. DC-B updates its UTDV entry for DC-A to 175.

4. We then revert DC-A back to its snapshot, and the highest committed USN returns to 100.
    1. This time, DC-A also resets its invocation ID and releases the RID pool because a change in VMGenID was detected.
    1. The AD database has returned to its state at the time of the snapshot but with a new invocation ID. The change in invocation ID notifies DC-B that DC-A has been restored, so DC-A can receive changes that may have been made and replicated after the snapshot was taken but before reverting to the snapshot.
5. We create 15 more users on DC-A.
    1. DC-B detects that DC-A has a new invocation ID, so it replicates those changes. DC-B adds an entry to the UTDV for DC-A's new invocation ID, with a USN of 115.
    1. DC-A replicates the original 75 users from its previous invocation ID.
6. The domain converges, replication is healthy, and all 90 users can authenticate against the domain as expected.

Maintain a Healthy AD Environment

Introducing virtualization to AD DCs provides many benefits but also raises challenges, with a potential for USN rollback and SID duplication. Microsoft worked with hypervisor manufacturers, including their own Hyper-V hypervisor, to enhance AD and enable support for virtualized DCs by creating the VMGenID attribute. The VMGenID attribute is still essential today and will continue to be necessary to maintain a healthy AD environment as long as virtualized DCs exist.

Need help managing your Active Directory implementation to ensure a healthy environment? Our experts are here for you. Contact us today!