FalconFriday — Detecting Active Directory Data Collection — 0xFF21

When attackers gain access to a large corporate environment, one of the things they tend to do is extract large quantities of data from Active Directory. The extracted data can be analyzed using tools to find complex paths that allow privilege escalation and lateral movement.

Popular tools to collect data from Active Directory are:

- SharpHound, which is provided as part of BloodHound and is intended to effectively collect large quantities of data from an Active Directory environment

Windows-LDAP-Client'. LDAP queries are logged to the DeviceEvents table using the 'LdapSearch' action type.

The screenshot below shows an example of an LdapSearch event.

- SharpHound uses various distinctive patterns, for example, (|(samaccounttype=268435456)(samaccounttype=268435457)(samaccounttype=536870912)(samaccounttype=536870913))

Knowing this, we can write a detection that searches for these IOCs in the LDAP search filter.

The query is available on our Github account.

Detection method 2 — Domain controller LDAP query logging via Microsoft Defender for Identity

Microsoft Defender for Identity (MDI) provides a log of LDAP queries being executed against the domain controller in the IdentityQueryEvents table that can be queried via Advanced Hunting. LDAP queries are logged using the 'LDAP query' action type.

Below is an example of an LDAP query logged via MDI.

The LDAP search filters are also available from this event in the Query attribute. The collection tools can be detected using the same IoCs obtained before.

The query is available on our Github account.

Below is a list of pros and cons for this way of detecting the technique:

- Pro — Detects attacks on the domain controller side, and does not rely on monitoring of the endpoint from which the attack is executed

In Windows ,a SACL can be used to set up fine-grained access control and auditing on any so called 'securable object'. Since Active Directory entries are considered 'securable objects', an ACE that records access can be attached to them.

A nice resource which provides example ACE entries for SACLs, as well as a tool to programatically configure ACE and SACLs, is available in the Set-AuditRule repository from the OTRF project.

For this detection rule we will add 2 SACLs to log all read access to Users

When complete, there should be three Audit ACEs:

From now on, every time anyone reads the properties of a User, Group or Computer object, this will trigger a log entry.

Note that this can cause a significant amount of logging. It is recommended

To detect large-scale data collection from Active Directory, we can now write a query that looks at the number of 4662 events triggered for objects of type User, Computer and Group, by a specific user, in a given timeframe.

Since there might be users in the organization that need to query large amounts of information from Active Directory, some filtering might be required. One way that we have implemented this rule in the past, is by using a baseline approach where we look at the number of AD objects accessed by a user historically over the last 14 days. When the number of objects accessed on a given day is significantly larger than the historic

As mentioned at the start of this post, there are several methods to detect Active Directory data collection. Each method has its own pros and cons; in most cases a combination of these methods would be most successful.

. . .

Want to have access to our repository with over 350 advanced detections? Please have a look at our commercial offering and reach out via info@falconforce.nl.