

SAMRi10 Tool

Research Conducted by: Itai Grady, MicrosoftATA¹ Research Team

Written by: Itai Grady, MicrosoftATA Research Team

Reviewed by: Tal Be'ery, MicrosoftATA Research Team

October 2016

Table of Contents

1	Summary	2
2	Introduction	2
3	Security Account Manager (SAM) and Active Directory.....	3
4	Local Domains and Account database.....	3
5	Network Domains and Domain Controllers.....	3
5.1	SAMR: Remote Querying of SAM	3
5.1.1	Flow and Usage	3
6	SAMR Required Permissions.....	5
7	SAMRi10 details.....	6
7.1	Using SAMRi10.ps1.....	6
7.1.1	Revert Option	6
7.2	Results on SAMRi10 Hardened Targets	6
7.2.1	Net User on a Hardened Domain Controller	6
7.2.2	Get-NetLocalGroup Against a Hardened Machine.....	8

¹ <https://www.microsoft.com/en-us/cloud-platform/advanced-threat-analytics>

1 Summary

Reconnaissance (recon for short) is a key stage within the Advanced Attackers' kill chain. Once attackers have breached a single end-point, they need to discover their next targets within the victim's corporate network, most notably privileged users. In order to enable admins to harden their network against such recon attacks targeting local users, we had developed the "SAMRi10" (pronounced Samaritan) tool.

2 Introduction

Reconnaissance (recon for short) is a key stage within the Advanced Attackers' kill chain. Once attackers have breached a single end-point, they need to discover their next targets within the victim's corporate network, most notably privileged users

Attackers utilize compromised credentials in order to move laterally within their victims' network. These compromised credentials may consist of either domain or local credentials. Local credentials, especially those of local admins, are a lucrative target for the attackers as they are less managed (password complexity and change policy) and less monitored (no traffic and logs besides the specific computer).

Querying the Windows Security Account Manager (SAM) remotely via the SAM-Remote (SAMR) protocol against their victim's domain machines, allows the attackers to get all domain and local users with their group membership and map possible routes within the victim's network. Recently, some frameworks (e.g. BloodHound²) have automated that mapping process³.

By default, the SAM can be accessed remotely (via SAMR) by any authenticated user, including network connected users, which effectively means that any domain user is able to access it. Windows 10 had introduced an option to control the remote access to the SAM, through a specific registry value. On Windows Anniversary update (Windows 10 Version 1607⁴) the default permissions were changed to allow remote access only to administrators. An accompanying Group Policy setting was added, which gives a user-friendly interface to alter these default permissions.

In order to enable admins to have granular control over remote access to SAM for all Windows 10 versions, we had developed the "SAMRi10" (pronounced Samaritan) tool. The SAMRi10 tool is a short PowerShell (PS) script which alters these default permissions on all Windows 10 versions and Windows Server 2016. Most significantly, this hardening process should block attackers from easily getting valuable recon information.

² <https://github.com/adaptivethreat/BloodHound>

³ <https://www.blackhat.com/docs/eu-16/materials/eu-16-Beery-Grady-Cyber-Judo-Offensive-Cyber-Defense.pdf>

⁴ <https://support.microsoft.com/en-us/help/12387/windows-10-update-history>

3 Security Account Manager (SAM) and Active Directory

Accounts are always created relative to an issuing authority. In Windows, the issuing authority is referred to as a domain. A domain can be either a local domain or extend across a network. Domains store information about their accounts in an account database. Windows uses Active Directory as the account database in domain-based environments, whereas in environments that are not domain-based, it uses the security account manager (SAM) built-in database as the account database.⁵

4 Local Domains and Account database

Every computer that runs Windows has its own local domain, that is, it has an account database for accounts that are specific to that computer. These are referred to as local accounts, local groups, and so on. Because computers typically do not trust each other for account information, these identities stay local to the computer on which they were created.

5 Network Domains and Domain Controllers

In a network domain, certain Windows servers can be configured to be domain controllers. A domain controller is a server that has made its account database available to other machines in a controlled manner.

5.1 SAMR: Remote Querying of SAM

The Security Account Manager Remote Protocol (SAMR)⁶ exposes the security accounts manager database for a remote authenticated domain user. It does so for both **local** and **domain** accounts. There are five objects that are exposed by the protocol; server, domain, group, alias⁷ and user. All these objects can be updated and read, and some (user, group and alias) can also be created and deleted.

5.1.1 Flow and Usage

The basic flow of using the SAMR protocol is as such:

1. Connect to a server (the remote machine).
2. Enumerate/lookup the server for domains.
3. Open the domain of interest.
4. Lookup a user or alias/group in the domain.
5. Open the user/alias of interest.
6. Query the user/alias of interest.

⁵ <https://msdn.microsoft.com/en-us/library/gg604662.aspx>

⁶ <https://msdn.microsoft.com/en-us/library/cc245476.aspx>

⁷ For the precise definitions, see https://msdn.microsoft.com/en-us/library/cc223126.aspx#gt_f46053d6-0708-4094-ac63-57c1bcb73d32 and <https://msdn.microsoft.com/en-us/library/cc245661.aspx>. In short, alias can have members that are not part of its domain (whereas a group cannot).

There are a few tools that utilize these API calls, such as Net User⁸/Group⁹, PowerSploit's¹⁰ Get-NetLocalGroup¹¹ and Impacket's SAMRdump¹². Net User and Net Group are Windows built-in command line tools. With these tools an authenticated user can add or modify and display information on users or groups respectively on the local machine or its domain controller. The Get-NetLocalGroup queries a remote machine for its local groups (including the "Administrators" and "Users" groups). SAMRdump, queries the target machine for its local users (using the EnumDomainUsers on the target machine).

MicrosoftATA¹³ detects the use of such query and alerts the security administrator about it

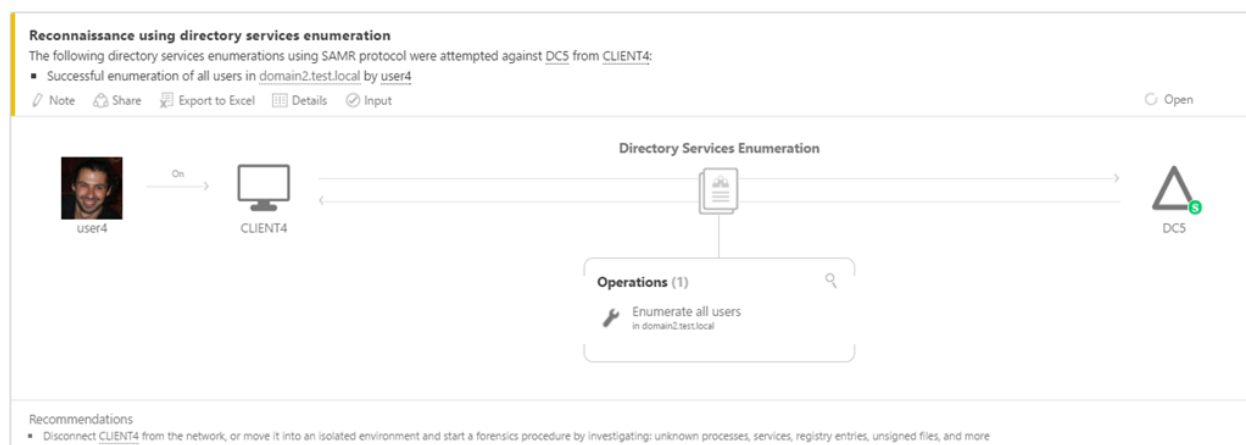


Figure 1 MicrosoftATA alert on Domain Users recon

⁸ [https://technet.microsoft.com/en-us/library/cc771865\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/cc771865(v=ws.11).aspx)

⁹ [https://technet.microsoft.com/en-us/library/cc754051\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/cc754051(v=ws.11).aspx)

¹⁰ <https://github.com/PowerShellMafia/PowerSploit>

¹¹ <https://github.com/PowerShellMafia/PowerSploit/blob/master/Recon/PowerView.ps1>

¹² <https://github.com/CoreSecurity/impacket/blob/master/examples/samrdump.py>

¹³ <https://www.microsoft.com/en-us/cloud-platform/advanced-threat-analytics>

6 SAMR Required Permissions

Prior to Windows 10, any domain user could query any computer for its local users via the SAMR protocol. In Windows 10, SAM remote permissions can be configured by setting the following registry value:

HKLM/System/CurrentControlSet/Control/Lsa/RestrictRemoteSAM

The Windows Anniversary update version changed the default security descriptor for the SAM access to limit the remote querying of SAM to local administrators¹⁴ only, **even if the aforementioned registry key is not present**, and added a Group Policy setting ("Network Access: Restrict clients allowed to make remote calls to SAM") to allow the central administration of this policy setting.

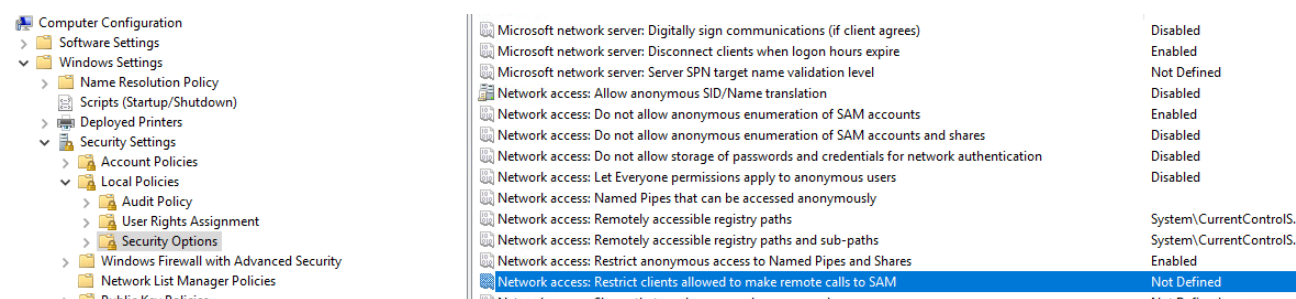


Figure 2: New Group Policy settings in anniversary update

Windows version	Who can query local users by default	Can default be changed
< Windows 10	Any domain user	No
Windows 10	Any domain user	Yes (only via registry)
> Windows10 (e.g. anniversary)	Only local administrators	Yes (registry or GPO)

RestrictRemoteSAM value is a string format of a Security Descriptor Definition Language (SDDL) which contains a Discretionary Access Control List (DACL) with a suitable Access Control Entry for allowed/denied users/groups.

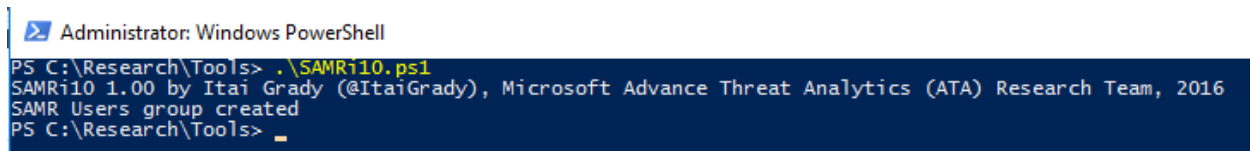
¹⁴ Security Identifier (Sid) S-1-5-32-544

7 SAMRi10 details

The SAMRi10 script hardens the remote access to the SAM by giving permission for members of Administrators group or the newly created group (also by this script) named "Remote SAM Users". This will allow any administrator or any service/user account added to the "Remote SAM Users" local group to remotely access SAM on the hardened machine.

7.1 Using SAMRi10.ps1

Run The SAMRi10 PowerShell script as administrator on the machine you wish to harden (Windows 10/Server 2016+).



```
Administrator: Windows PowerShell
PS C:\Research\Tools> .\SAMRi10.ps1
SAMRi10 1.00 by Itai Grady (@ItaiGrady), Microsoft Advance Threat Analytics (ATA) Research Team, 2016
SAMR Users group created
PS C:\Research\Tools> _
```

Figure 3: \SAMRi10.ps1 execution

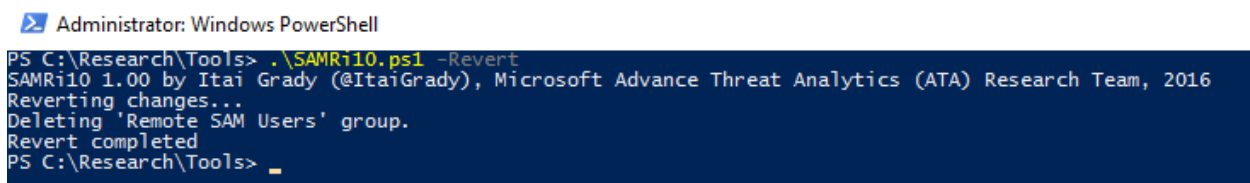
To allow Service/User account to remotely access SAM on the hardened machine, please add it to the newly created "Remote SAM Users" group. (as seen in Figure 8)

7.1.1 Revert Option

To revert changes done by the SAMRi10 tool, use the Revert option.

Registry value will be set to the backed up value and the "Remote SAM Users" group will be deleted.

For example:



```
Administrator: Windows PowerShell
PS C:\Research\Tools> .\SAMRi10.ps1 -Revert
SAMRi10 1.00 by Itai Grady (@ItaiGrady), Microsoft Advance Threat Analytics (ATA) Research Team, 2016
Reverting changes...
Deleting 'Remote SAM Users' group.
Revert completed
PS C:\Research\Tools> _
```

Figure 4: \SAMRi10.ps1 -Revert

7.2 Results on SAMRi10 Hardened Targets

7.2.1 Net User on a Hardened Domain Controller

A Windows Server 2016 domain controller, hardened by the SAMRi10 tool, will respond differently to a remote SAM access, based upon the requesting user account type:

- Domain Admin account: Querying a hardened domain controller, with the "Net User/Group" for example, will be completed successfully.
- Non-privileged User account: Querying a hardened domain controller, with the "Net User/Group" for example, will result with an "Access is denied" error.

- Member of "Remote SAM Users": Querying a hardened domain controller, with the "Net User/Group" for example, will be completed successfully.

The following figures represent the scenarios described above:

```

C:\windows\system32\cmd.exe

c:\Tools>whoami
domain1\administrator

c:\Tools>net user /domain
The request will be processed at a domain controller for domain domain1.test.local.

User accounts for \\2016-DC1.domain1.test.local

-----
ABanks          ABarstow        ABauer
ACupar          ADanford        Administrator
ADuran          AHouston        APendleton
AReynolds       AShiel          AWaters
AWilkinson     BArmstrong      BBalcombe
BBaldwin       BBallantyne     BBarton
BBellamy       BBlaney         BBogue
BBrocklesby    BBryan          BBuckner
BBuxton        BCarpenter      BClayton
BCollamore     BColon         BCote
  
```

Figure 5:Administrator successfully calls Net User from remote on a hardened domain controller

```

Select Administrator: Command Prompt

c:\Tools>whoami
domain1\User2

c:\Tools>net user /domain
The request will be processed at a domain controller for domain domain1.test.local.

System error 5 has occurred.
Access is denied.

c:\Tools>_
  
```

Figure 6:User2 (non-admin) gets access denied calling Net User remotely to a hardened Domain Controller

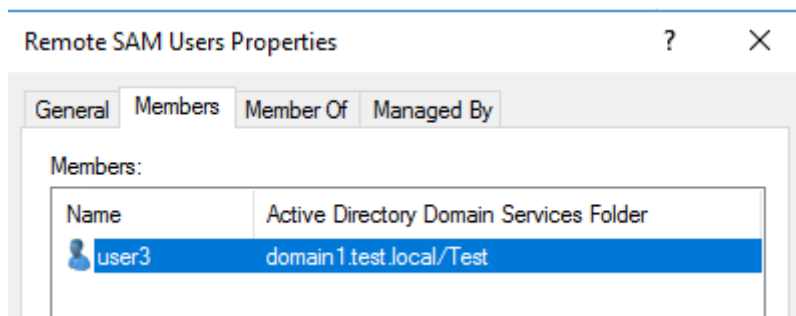


Figure 7:Group membership of Remote SAM Users on the hardened Domain Controller

```
C:\windows\system32\cmd.exe

c:\Tools>whoami
domain1\user3

c:\Tools>net user /domain
The request will be processed at a domain controller for domain domain1.test.local.

User accounts for \\2016-DC1.domain1.test.local

-----
ABanks          ABarstow        ABauer
ACupar          ADanford        Administrator
ADuran          AHouston        APendleton
AREynolds       AShiel          AWaters
AWilkinson      BArmstrong      BBalcombe
BBaldwin        BBallantyne     BBarton
BBellamy        BBlaney         BBogue
BBrocklesby     BBryan          BBuckner
BBuxton         BCarpenter      BClayton
BCollamore      BColon          BCote
BCranstoun      BCrosswell      BDole
BDuncan         BFinley         BFoley
BFrank          BFry            BGihon
```

Figure 8: User3 (non-admin, but member of "Remote SAM Users") successfully calls Net User on a hardened Domain Controller

7.2.2 Get-NetLocalGroup Against a Hardened Machine

A Windows 10 machine, hardened by the SAMRi10 tool, will respond to a remote SAM access, based upon the requesting user account type, similar to a hardened 2016 domain controller.

Remote execution of PowerSploit's Get-NetLocalGroup method against a SAMRi10 hardened computer, using an unprivileged user will result with an "Access is denied" error.

Executing the same method, with an administrative account or a member of the local "Remote SAM Users" on the remote machine, will be completed successfully.

The following figures represent the scenarios described above:

```
PS C:\Users> whoami
domain1\user2
PS C:\Users> Get-NetLocalGroup -ComputerName Client5
WARNING: [!] Error: Exception calling "Invoke" with "2" argument(s): "Access is denied."
PS C:\Users> _
```

Figure 9: user2 (non-admin) gets access denied call Get-NetLocalGroup on a hardened Windows 10 machine


```
PS C:\Users> whoami
domain1\user1
PS C:\Users> Get-NetLocalGroup -ComputerName Client5

Server      : Client5
AccountName : DOMAIN1/Client5/Administrator
SID         : S-1-5-21-1749101508-598534307-36209114-500
Disabled    : True
IsGroup     : False
IsDomain    : False
LastLogin   :

Server      : Client5
AccountName : DOMAIN1/Client5/user5
SID         : S-1-5-21-1749101508-598534307-36209114-1001
Disabled    : False
IsGroup     : False
IsDomain    : False
LastLogin   : 11/27/2016 9:56:57 AM

Server      : Client5
AccountName : domain1.test.local/Domain Admins
SID         : S-1-5-21-989687458-3461180213-172365591-512
Disabled    : False
IsGroup     : True
IsDomain    : True
LastLogin   :

PS C:\Users> _
```

Figure 10:user1 (local admin) successfully calls Get-NetLocalGroup on a hardened Windows 10 machine