Exploring Attack Paths Using Graph Theory: Case - Microsoft Entra ID Pass-Through Authentication

Nestori Syynimaa[a]
Principal Identity Researcher, Microsoft, Threat Intelligence Center,
Faculty of Information Technology, University of Jyväskylä, Jyväskylä, Finland

Keywords: Graph Theory, Entra ID, Attack Paths, Authentication.

Abstract: Graphs have been used to describe attack paths since the 1990s. They are powerful ways to present complex problems in a relatively simple way. Microsoft Entra ID is an identity and access management (IAM) solution most private and public sector organisations use. As an IAM, it supports multiple authentication methods. One little-researched authentication method is pass-through authentication (PTA). This paper presents the findings of a study researching PTA for novel vulnerabilities. The findings reveal vulnerabilities that enable novel PTA-related attacks, allowing threat actors to gain remote, persistent, and undetectable access to the target organisation's Entra ID. Threat actors could exploit these vulnerabilities to create backdoors, harvest credentials, and perform DoS attacks. The found attack paths were depicted in the PTA Attack Graph, which is the main contribution of this paper.

1 INTRODUCTION

Graphs are discrete structures consisting of nodes and edges connecting them. They can be used to solve problems of virtually any discipline, including cyber security (Dawood, 2014). Graphs have been used to describe attack paths since the 1990s (see Amoroso, 1994; Schneier, 1999; Weiss, 1991). In this paper, we will use graphs to describe different paths attackers can have from the start of the attack to the end of the attack. For this purpose, we adopted BPMN notation (OMG, 2011) to describe the *actions* threat actors need to take to perform the attack.

Entra ID is Microsoft's Identity and Access Management (IAM) service (Microsoft, 2024), used by most public and private sector organisations. Entra ID supports multiple authentication methods. One of the methods is pass-through authentication (PTA), which allows the same password to be used in on-premises Active Directory (AD) and Entra ID (Microsoft, 2023e).

In this paper, we seek an answer to the following research question: "What vulnerabilities exist in pass-through authentication?"

2 PREVIOUS RESEARCH

2.1 Pass-Through Authentication

The PTA functionality is illustrated in Figure 1. From the user's perspective, the process is as follows:

1. The user signs in on Entra ID
2. The user's credentials are sent to the PTA agent
3. The PTA agent attempts to sign in to Active Directory using the user's credentials

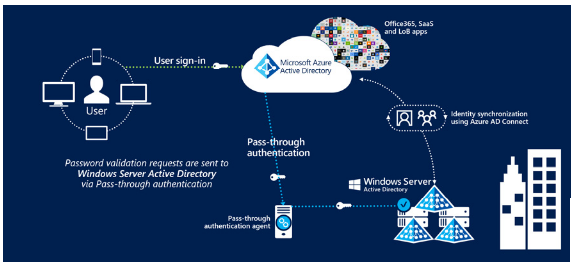

Office365, SaaS and LoB apps
User sign-in
Microsoft Azure Active Directory
User
Identity synchronization using Azure AD Connect
Pass-through authentication
Password validation requests are sent to Windows Server Active Directory via Pass-through authentication
Windows Server Active Directory
Pass-through authentication agent

Figure 1: Pass-Through Authentication (Microsoft, 2023e).

Microsoft has not published the technical details on how PTA works. However, many researchers have

[a] https://orcid.org/0000-0002-6848-094X

Syynimaa, N.
Exploring Attack Paths Using Graph Theory: Case - Microsoft Entra ID Pass-Through Authentication.
DOI: 10.5220/0013119100003899

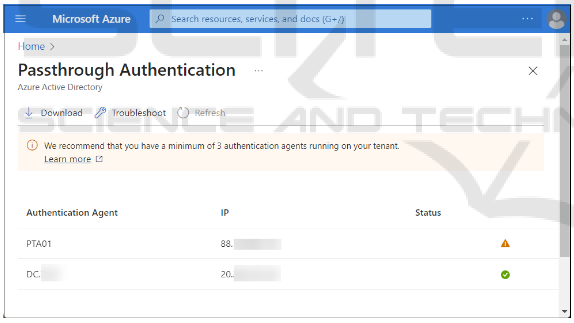
In *Proceedings of the 11th International Conference on Information Systems Security and Privacy (ICISSP 2025) - Volume 2*, pages 486-492
ISBN: 978-989-758-735-1; ISSN: 2184-4356

published their findings regarding PTA (see Chester, 2019; Felton, 2017; Syynimaa, 2022).

PTA agents use a certificate (stored in LocalMachine personal store) for decrypting authentication requests and certificate-based authentication (Felton, 2017). During its start-up process, the PTA agent retrieves a configuration file called *bootstrap* from Entra ID, which contains URLs for Azure Service Bus endpoints (Felton, 2017). Entra ID uses Azure Service Bus to send encrypted authentication requests to PTA agents. After receiving the authentication request, the PTA agent decrypts the requests, passes the included credentials to the *LogonUserW* function, and returns the result to Entra ID (Felton, 2017). The protocol details used by the PTA agent to communicate with Entra ID were published in 2020 (Syynimaa, 2020b).

2.2 Logging and Monitoring PTA Agents

When a PTA agent is registered to Entra ID during the installation, it appears in the Azure Portal. Administrators can see the IP address, the computer's name running the agent, and the agent status (active/inactive) (Figure 2).



Passthrough Authentication

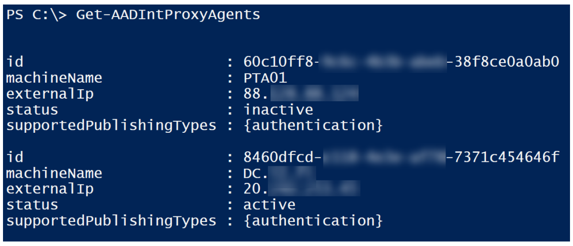
Azure Active Directory

We recommend that you have a minimum of 3 authentication agents running on your tenant.
Learn more

Authentication Agent	IP	Status
PTA01	88.[redacted]	⚠
DC.[redacted]	20.[redacted]	✓

Figure 2: List of PTA agents in Azure Portal.

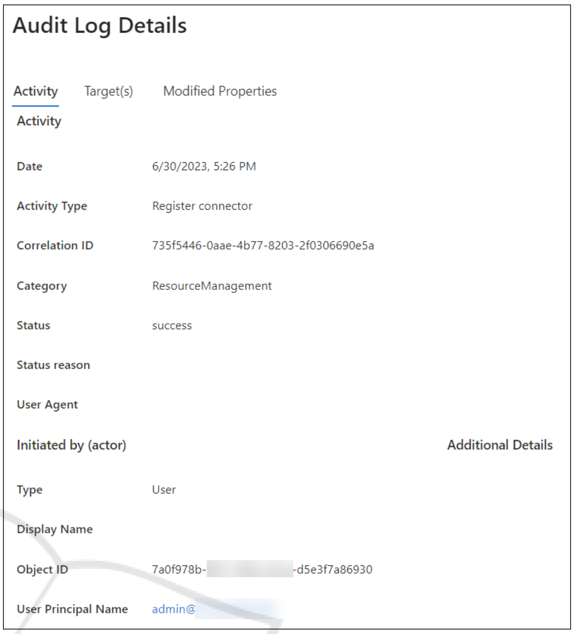
The PTA agent information can also be accessed programmatically, for instance, using the AADInternals toolkit (Figure 3). This also allows the viewing of the PTA Agent IDs that were not shown in the Azure Portal.



```
PS C:\> Get-AADIntProxyAgents

id                          : 60c10ff8-[redacted]-38f8ce0a0ab0
machineName                 : PTA01
externalIp                  : 88.[redacted]
status                      : inactive
supportedPublishingTypes    : {authentication}

id                          : 8460dfcd-[redacted]-7371c454646f
machineName                 : DC.[redacted]
externalIp                  : 20.[redacted]
status                      : active
supportedPublishingTypes    : {authentication}
```

Figure 3: List of PTA agents in AADInternals.

A *Register connector* event is added to the Entra ID *Audit log* during the registration. The event shows who registered the agent and when, but the agent ID is not shown (Figure 4).



Audit Log Details

Activity

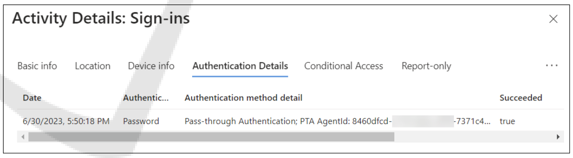
Date	6/30/2023, 5:26 PM
Activity Type	Register connector
Correlation ID	735f5446-0aae-4b77-8203-2f0306690e5a
Category	ResourceManagement
Status	success
Status reason	
User Agent	

Initiated by (actor) — Additional Details

Type	User
Display Name	
Object ID	7a0f978b-[redacted]-d5e3f7a86930
User Principal Name	admin@[redacted]

Figure 4: "Register connector" event in Entra ID Audit log.

Each time a user is authenticated using PTA, a corresponding event is added to the Entra ID *Sign-ins log*. On the *Authentication Details* tab, the PTA agent ID is shown (Figure 5).



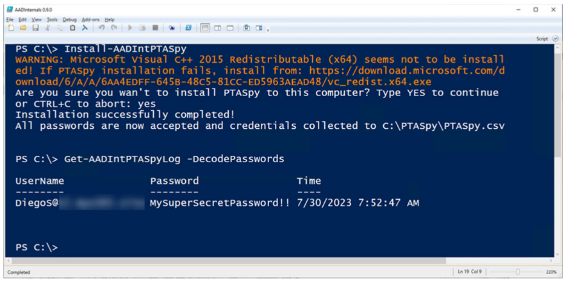
Activity Details: Sign-ins

Date	Authentic...	Authentication method detail	Succeeded
6/30/2023, 5:50:18 PM	Password	Pass-through Authentication; PTA AgentId: 8460dfcd-[redacted]-7371c4...	true

Figure 5: Authentication Details in Entra ID Sign-ins log.

2.3 Attacking Pass-Through Authentication

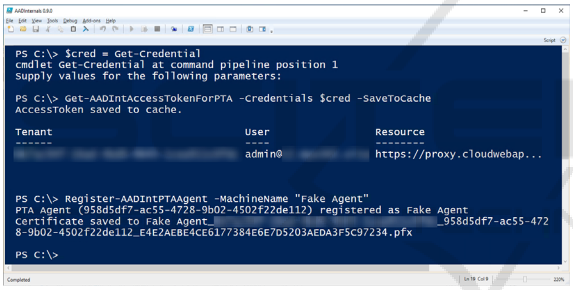
Chester (2019) introduced a novel PTA-related attack vector. After compromising a computer running a PTA agent, threat actors could replace the *LogonUserW* function by injecting a Dynamic Linking Library (DLL) into the PTA agent process. This allowed threat actors to (i) harvest credentials and (ii) allow or deny login requests. This technique required persistent access to the PTA agent, as restarting the agent would remove the injected DLL. Syynimaa (2021) published a PTASpy (Figure 6) leveraging the DLL injection attack introduced by Chester. PTASpy was included in the AADInternals toolkit in 2020. Kalendarov and Beber (2024) used the same technique in their recent report. As this attack takes place on a server running the PTA agent, it can be detected by monitoring the server.



```
PS C:\> Install-AADIntPTASpy
WARNING: Microsoft Visual C++ 2015 Redistributable (x64) seems not to be install
ed! If PTASpy installation fails, install from: https://download.microsoft.com/d
ownload/6/A/A/6AA4EDFF-645B-48C5-81CC-ED5963AEAD48/vc_redist.x64.exe
Are you sure you wan't to install PTASpy to this computer? Type YES to continue
or CTRL+C to abort: yes
Installation successfully completed!
All passwords are now accepted and credentials collected to C:\PTASpy\PTASpy.csv


PS C:\> Get-AADIntPTASpyLog -DecodePasswords

UserName              Password               Time
--------              --------               ----
DiegoS@[redacted] MySuperSecretPassword!! 7/30/2023 7:52:47 AM



PS C:\>
```

Figure 6: Installing PTASpy with AADInternals.

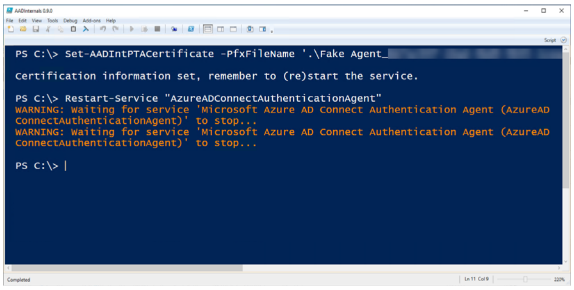
AADInternals allows registering "fake" PTA agents (Figure 7), which creates certificates that PTA agents can use. This requires that the threat actor has either *Global Administrator* or *Hybrid Identity Administrator* role in Entra ID. This attack can be detected as the new agent appears in the Azure Portal and registration is logged in the audit log.



```
PS C:\> $cred = Get-Credential
cmdlet Get-Credential at command pipeline position 1
Supply values for the following parameters:

PS C:\> Get-AADIntAccessTokenForPTA -Credentials $cred -SaveToCache
AccessToken saved to cache.

Tenant                                User              Resource
------                                ----              --------
[redacted]                            admin@[redacted]  https://proxy.cloudwebap...



PS C:\> Register-AADIntPTAAgent -MachineName "Fake Agent"
PTA Agent (958d5df7-ac55-4728-9b02-4502f22de112) registered as Fake Agent
Certificate saved to Fake Agent_[redacted]_958d5df7-ac55-472
8-9b02-4502f22de112_E4E2AEBE4CE6177384E6E7D5203AEDA3F5C97234.pfx

PS C:\>
```

Figure 7: Registering fake PTA agents.

Existing PTA agents can configured to use the newly created certificate (Figure 8). The PTA agent can be installed on a computer under control of threat actor. This means that configuration can not be detected by the target organisation. However, the usage of the PTA agent can be detected from the sign-ins log.



```
PS C:\> Set-AADIntPTACertificate -PfxFileName '.\Fake Agent_[redacted]

Certification information set, remember to (re)start the service.

PS C:\> Restart-Service "AzureADConnectAuthenticationAgent"
WARNING: Waiting for service 'Microsoft Azure AD Connect Authentication Agent (AzureAD
ConnectAuthenticationAgent)' to stop...
WARNING: Waiting for service 'Microsoft Azure AD Connect Authentication Agent (AzureAD
ConnectAuthenticationAgent)' to stop...

PS C:\> |
```

Figure 8: Configuring the PTA agent to use the provided certificate.

The PTA Attack Graph in Figure 10 depicts the known PTA-related attacks. The lower path requires local admin privileges to the computer running an PTA agent. If the computer is compromised, a threat actor can install PTASpy (or other similar tools). The upper path requires one of the mentioned roles for to target organisation's Entra ID. If the threat actor has either of the roles, a fake PTA agent can be registered. After that, an existing threat actor controlled PTA agent can be configured to use the resulting certificate. As the computer is under control of threat actor, PTASpy can be installed to start gathering credentials and accept all passwords.

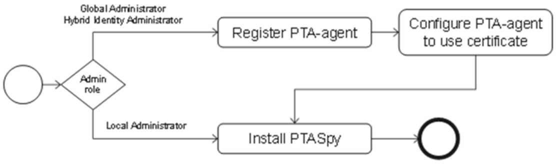


Figure 9: PTA Attack Graph v1.

3 RESEARCH METHODOLOGY

The research approach should be selected according to the aim of the research (Järvinen, 2018). In this research, the research aim is two-folded. First, we are studying the reality, *i.e.*, what vulnerabilities exist in PTA. Second, we produce a graph to describe attack paths enabled by the vulnerabilities found. Following the research approach taxonomy by Järvinen (2018), we will use *theory-creating* approaches for the former research aim and *innovation-building* approaches for the latter.

Theory-creating research aims to increase understanding of the research subject. In this research, we will first study how PTA behaves in normal operation to identify possible vulnerabilities. This can be categorised as *descriptive observational* research (Edgar & Manz, 2017). To study the PTA behaviour, we adopted *the reverse-engineering method,* commonly used to study existing systems (Eilam, 2005). We used two tools for this purpose. First, we used *Fiddler* (Telerik, 2023) to monitor the traffic between the PTA agent and Entra ID. Second, we used Process Monitor (Microsoft, 2023d), *ProcMon* for short, to monitor real-time filesystem and registry activity. The research setup is illustrated in Figure 10.

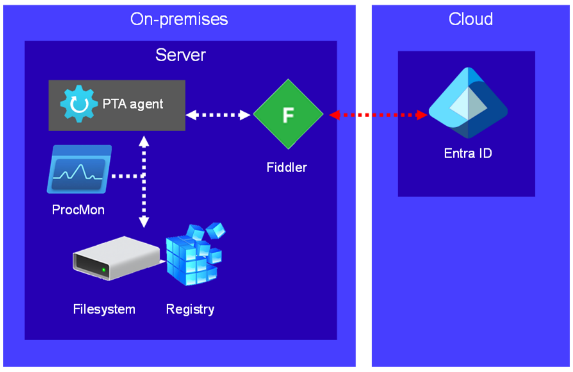


Figure 10: Research setup.

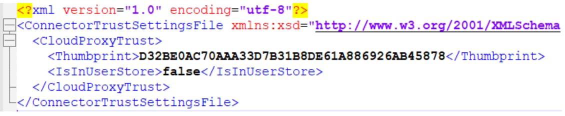
The innovation-building research aims to create an artefact to answer the research question. In this research, we produce a graph that depicts the vulnerabilities found during the theory-creating research. As such, this is a Design Science Research (DSR) and thus follows the Design Science Research Methodology (DSRM) by Peffers *et al.* (2007).

4 RESULTS

4.1 Exporting PTA Agent Certificate

On Windows computers, the certificates can be exported from the certificate store with Certificate Manager. Private keys can only be exported for certificates that were marked *exportable* at the time of importing the certificate. A technique to export the private key regardless of the exportability status was introduced in early 2022 (Syynimaa, 2022).

The thumbprint of the certificate used by the PTA agent is stored in a configuration file (see Figure 11). The certificate with the public key is stored in the Local Computer Personal store when *IsInUserStore* setting is *false*. This is the case after installing the PTA agent. When *true*, the certificate is stored in Personal store of *Network Service*. This is because PTA agent renews the certificate every 30 days and Network Service doesn't have access to the Local Machine store (Microsoft, 2023a). The certificate includes the name of the private key file, which is stored on disk and encrypted with the Data Protection Application Protection Interface (DPAPI). DPAPI is an encryption and decryption API inclued in the .NET framework (Microsoft, 2023c).



```
<?xml version="1.0" encoding="utf-8"?>
<ConnectorTrustSettingsFile xmlns:xsd="http://www.w3.org/2001/XMLSchema
  <CloudProxyTrust>
    <Thumbprint>D32BE0AC70AAA33D7B31B8DE61A886926AB45878</Thumbprint>
    <IsInUserStore>false</IsInUserStore>
  </CloudProxyTrust>
</ConnectorTrustSettingsFile>
```

Figure 11: PTA Agent configuration file.

Although the file name of the private key was known, the location on disk for Network Service was not. The location was revealed using ProcMon to see what files the PTA agent service was accessing. That allowed exporting the certificate and private key regardless of where they were stored.

4.2 Exploiting Certificate Using Microsoft PTA Agent

Using an exported certificate in a threat actor-controlled PTA agent differs from registering a fake agent. Exporting the certificate takes place on the computer running the PTA agent, so there won't be any log entries in Entra ID.

For high availability, at least three PTA agents should be installed on servers joined to the same Active Directory (AD) (Microsoft, 2023b). However, threat actor-controlled servers are not members of the victim's AD. If a threat actor-controlled PTA agent is configured to use the exported certificate and PTASpy is installed, any username and password combination is accepted and stored in a file. If PTASpy is not installed, all authentication requests will fail as victims' credentials are not present in threat actor-controlled servers. This allows threat actors to use PTA agents for Denial-of-Service (DoS) attacks.

When using an exported certificate on a threat actor-controlled server, the PTA agent name was changed on Azure Portal to match the threat actor-controlled server. Exploitation could be detected by monitoring PTA agent name changes unless the name of the threat actor-controlled server was changed to match the victim's server.

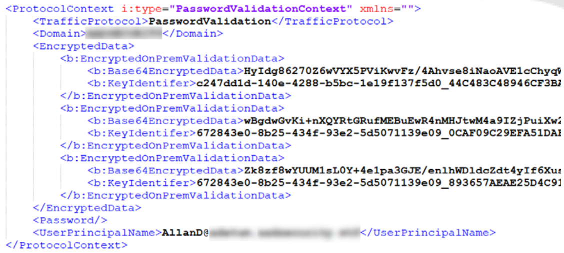
The challenge with exploiting exported certificates is the lack of persistence, as the certificate needs to be renewed every 30 days. However, it turned out that both the original and exported certificate could be used to acquire new valid certificate. Both certificates had the same agent identity information but different thumbprints. As such, there can be multiple certificates per PTA agent at any given time. Moreover, the certificate renewal is not logged in Entra ID Audit log.

4.3 Exploiting Certificate Using a Custom PTA Agent

The initial Proof-of-Concept (PoC) of a custom PTA agent was published on March 2020 (Syynimaa, 2020a). The PoC was further developed into a full-blown offensive tool during the research. The details of the PoC are out of this paper's scope, but the key findings are as follows.

The first observation was that the PTA agent's IP address (and name) changed on Azure Portal only when the bootstrap was fetched. That happens during the agent start-up and then once every five minutes. Configuring the custom agent to use existing bootstrap allows threat actors to exploit exported certificates silently. The bootstrap could be exported at the same time as exporting the PTA agent certificate.

The second observation was that the authentication request contains the credentials of the user trying to log in to Entra ID. The credentials are encrypted, so they can only be decrypted by a corresponding certificate. As PTA supports multiple agents, the authentication requests contain user's credentials encrypted with the public keys of all agents. As we noticed earlier, one agent can have multiple certificates. It turns out that the authentication request contains encrypted entries for all certificates. The format of the key identifiers seen in Figure 12 is "<AgentId>_<CertificateThumbprint>". As we can see, the agent with ID starting "672843e0" has two entries with different thumbprints.



```
<ProtocolContext i:type="PasswordValidationContext" xmlns="">
    <TrafficProtocol>PasswordValidation</TrafficProtocol>
    <Domain>[redacted]</Domain>
    <EncryptedData>
        <b:EncryptedOnPremValidationData>
            <b:Base64EncryptedData>HyIdg86270Z6wVYX5PViKwvFz/4Ahvse8iNaoAVE1cChyq
            <b:KeyIdentifer>c247dd1d-140e-4288-b5bc-1e19f137f5d0_44C483C48946CF3B
        </b:EncryptedOnPremValidationData>
        <b:EncryptedOnPremValidationData>
            <b:Base64EncryptedData>wBgdwGvKi+nXQYRtGRufMEBuEwR4nMHJtwM4a9IZjPuiXw
            <b:KeyIdentifer>672843e0-8b25-434f-93e2-5d5071139e09_0CAF09C29EFA51DA
        </b:EncryptedOnPremValidationData>
        <b:EncryptedOnPremValidationData>
            <b:Base64EncryptedData>Zk8zf8wYUUM1sL0Y+4e1pa3GJE/en1hWD1dcZdt4yIf6Xu
            <b:KeyIdentifer>672843e0-8b25-434f-93e2-5d5071139e09_893657AEAE25D4C9
        </b:EncryptedOnPremValidationData>
    </EncryptedData>
    <Password/>
    <UserPrincipalName>AllanD@[redacted]</UserPrincipalName>
</ProtocolContext>
```

Figure 12: Content of PTA authentication request (adapted from Secureworks, 2022).

The third observation was that the maximum number of PTA agents is 40 (Microsoft, 2023b), but there is no public information about the maximum number of certificates per agent. After experimenting with renewing certificates multiple times, the limit appears to be ten certificates per agent. Further, certificates seemed to work with the FIFO principle, meaning that the entries for the older certificates were dropped from the authentication requests when certificates were renewed. This allows attackers to perform a new kind of DoS attack, as they could renew certificates until the original PTA agent certificate is dropped from the authentication requests. This would effectively prevent the original PTA agent from decrypting and handling authentication requests.

The custom PTA agent can also be configured to return arbitrary Windows error messages to Entra ID, enabling DoS attacks.

4.4 Summary

4.4.1 Findings

Vulnerabilities enabling novel PTA-related attacks were found during the research. These vulnerabilities allow threat actors to gain remote, persistent, and undetectable access to the target organisation Entra ID.

First, after successfully compromising the victim PTA agent server, a threat actor can export the PTA agent certificate and bootstrap. These can then be used on threat actor-controlled PTA agent servers. This, in turn, allows threat actors to harvest credentials, create a backdoor, or perform DoS attacks.

Second, threat actors can renew the certificate indefinitely, enabling permanent access to the victim's Entra ID.

Third, only the registration of PTA agents are logged in the Entra ID Audit log. In the Sign-ins log, the agent ID performing the authentication is shown, but not the used certificate. Thus, exploitation of exported PTA agent certificates can not be detected.

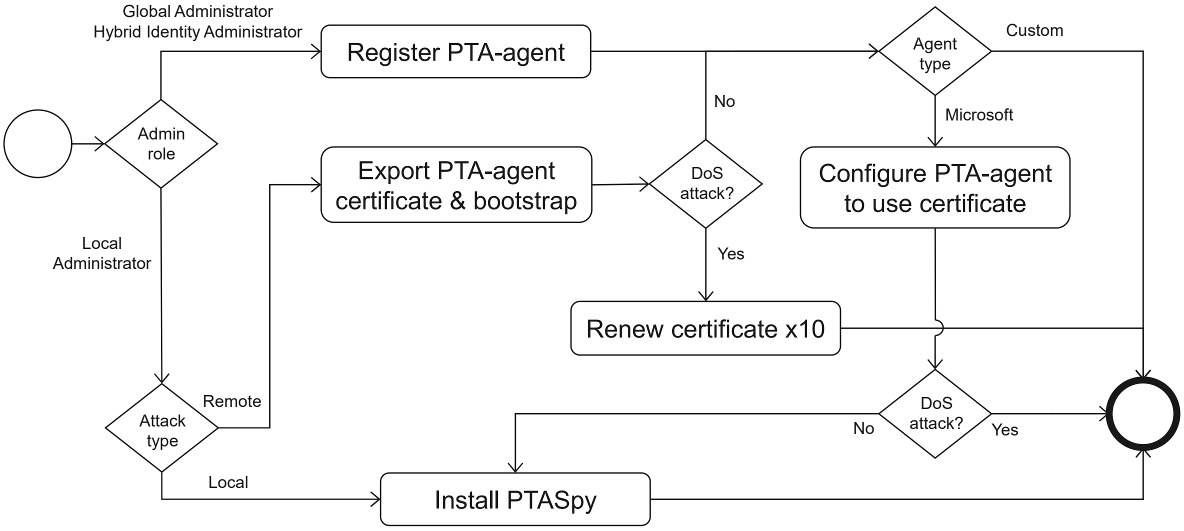


Figure 13: PTA Attack Graph v2.

4.4.2 PTA Attack Graph

The final PTA attack graph is depicted in Figure 13. There is a new port for attack type if the threat actor has local administrator permissions to the server running PTA agent. The *local* attack involves installing a PTASpy on the victim's PTA agent. The *remote* attack starts by exporting the PTA agent certificate and bootstrap. After exporting, if the threat actor desires to use the certificate for DoS attacks, it can be renewed ten times. If not, the threat actor can use the certificate with a custom agent or configure Microsoft PTA agent to use it. If latter, it can be used for a DoS attack or as a backdoor and credential harvester by installing PTASpy.

5 DISCUSSION

5.1 Research Question and Aims

The research question of this paper was "What vulnerabilities exist in pass-through authentication?". Two aims were set to answer the research question. The first one was to research PTA to find vulnerabilities. The second one was to depict the findings in an easy-to-understand attack graph. Both aims were achieved, and thus, an answer to the research question was provided.

5.2 Implications

The research has both scientific and practical implications.

This research introduced a PTA Attack Graph depicting current knowledge of PTA-related attacks. As such, it expands the scientific understanding of PTA-related cybersecurity research area.

The PTA Attack Graph increases the general public's knowledge of PTA security issues. This also helps defend against PTA-related attacks.

5.3 Future Work

Future work should focus on studying how the exploitation of exported PTA agent certificates could be detected besides Entra ID Sign-ins and Audit logs.

5.4 Research Rigour

The resulting artefact of DSR should be evaluated to demonstrate its utility and quality (Baskerville, Kaul, & Storey, 2017). The PTA Attack Graph can be categorised as a model. The model's validity is revealed when it confronts empirical facts (Barlas, 1996). The attacks depicted in the graph emerged during the empirical research and were confirmed by Microsoft. As such, it can be stated that the PTA Attack Graph is valid, *i.e.*, it models the current knowledge of PTA-related attacks.

5.5 Acknowledgement

The research findings were reported to Microsoft, but the vulnerabilities found were not seen as posing additional risks as the new attacks would require local administrator permissions.

REFERENCES

Amoroso, E. G. (1994). *Fundamentals of Computer Security Technology*. USA: Prentice-Hall.

Barlas, Yaman. (1996). Formal aspects of model validity and validation in system dynamics. *System Dynamics Review, 12*(3), 183-210.

Baskerville, Richard, Kaul, Mala, & Storey, Veda. (2017). *Establishing Reliability in Design Science Research*.

Chester, Adam. (2019). Azure AD Connect for Red Teamers. Retrieved from https://blog.xpnsec.com/azuread-connect-for-redteam/

Dawood, Harith A. (2014, 29-30 Dec. 2014). *Graph Theory and Cyber Security.* Paper presented at the 2014 3rd International Conference on Advanced Computer Science Applications and Technologies.

Edgar, Thomas W., & Manz, David O. (2017). *Research Methods for Cyber Security*. Cambridge, MA, United States: Syngress.

Eilam, Eldad. (2005). *Reversing: secrets of reverse engineering*: John Wiley & Sons.

Felton, Matt. (2017). Azure AD Pass-through Authentication – How does it work? Part 2. Retrieved from https://journeyofthegeek.com/tag/azure-pass-through-authentication/

Järvinen, Pertti. (2018). *On Research Methods*. Retrieved from https://learning2.uta.fi/pluginfile.php/712390/mod_resource/content/4/On%20research%20methods.pdf

Kalendarov, Ilan, & Beber, Elad. (2024). Double Agent: Exploiting Pass-through Authentication Credential Validation in Azure AD Retrieved from https://cymulate.com/blog/exploiting-pta-credential-validation-in-azure-ad/

Microsoft. (2023a). Azure Active Directory pass-through authentication security deep dive. Retrieved from https://learn.microsoft.com/en-us/azure/active-directory/hybrid/connect/how-to-connect-pta-security-deep-dive

Microsoft. (2023b). Azure Active Directory Pass-through Authentication: Quickstart. Retrieved from https://learn.microsoft.com/en-us/azure/active-directory/hybrid/connect/how-to-connect-pta-quick-start

Microsoft. (2023c). How to: Use Data Protection. Retrieved from https://learn.microsoft.com/en-us/dotnet/standard/security/how-to-use-data-protection

Microsoft. (2023d). Process Monitor. Retrieved from https://learn.microsoft.com/en-us/sysinternals/downloads/procmon

Microsoft. (2023e). User sign-in with Microsoft Entra pass-through authentication. Retrieved from https://learn.microsoft.com/en-us/entra/fundamentals/whatis

Microsoft. (2024). What is Microsoft Entra ID? Retrieved from https://learn.microsoft.com/en-us/entra/fundamentals/whatis

OMG. (2011). Documents Associated with Business Process Model and Notation (BPMN) Version 2.0. Retrieved from http://www.omg.org/spec/BPMN/2.0/

Peffers, Ken, Tuunanen, Tuure, Rothenberger, Marcus A, & Chatterjee, Samir. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of management information systems*, 45-77.

Schneier, Bruce. (1999). Attack trees: modeling security threats. *Dr. Dobb's Journal, 24*(12), 21,29. Retrieved from https://www.drdobbs.com/attack-trees/184411129

Secureworks. (2022). Azure Active Directory Pass-Through Authentication Flaws. Retrieved from https://www.secureworks.com/research/azure-active-directory-pass-through-authentication-flaws

Syynimaa, Nestori. (2020a). AADInternals. PTAAgent.cs sourcecode. Retrieved from https://github.com/Gerenios/AADInternals/blob/073c9511b5d8d42795e26ccbab1d07e9c5cf95a6/PTAAgent.cs

Syynimaa, Nestori. (2020b). Deep-dive to Azure AD Pass-Through Authentication. Retrieved from https://aadinternals.com/post/pta-deepdive

Syynimaa, Nestori. (2021). PTASpy sourcecode. Retrieved from https://github.com/Gerenios/public/blob/master/PTASpy.cpp

Syynimaa, Nestori. (2022). Stealing and faking Azure AD device identities. Retrieved from https://aadinternals.com/post/deviceidentity/

Telerik. (2023). Fiddler Overview. Retrieved from https://www.telerik.com/fiddler

Weiss, Jonathan D. (1991). *A System Security Engineering Process.* Paper presented at the 14th National Computer Security Conference (NIST/NCSC), Washington.