

# Attack Flow Training: 2 – Using Attack Flow Builder

Online Training



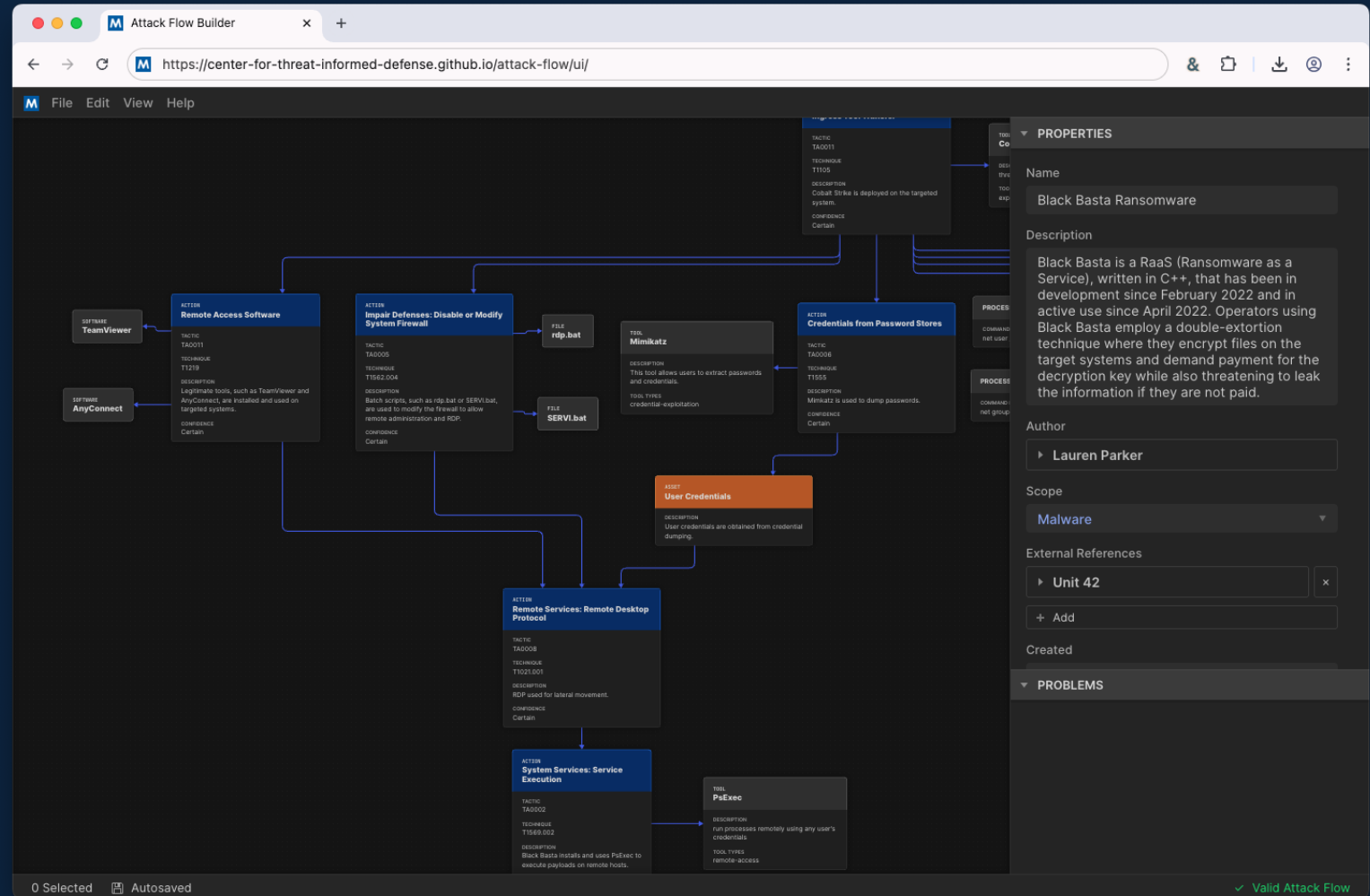
# Agenda

- 1 – Introduction to Attack Flow
- **2 – Using Attack Flow Builder**
- 3 – Building An Attack Flow
- 4 – Visualization
- 5 – What's New in V3?

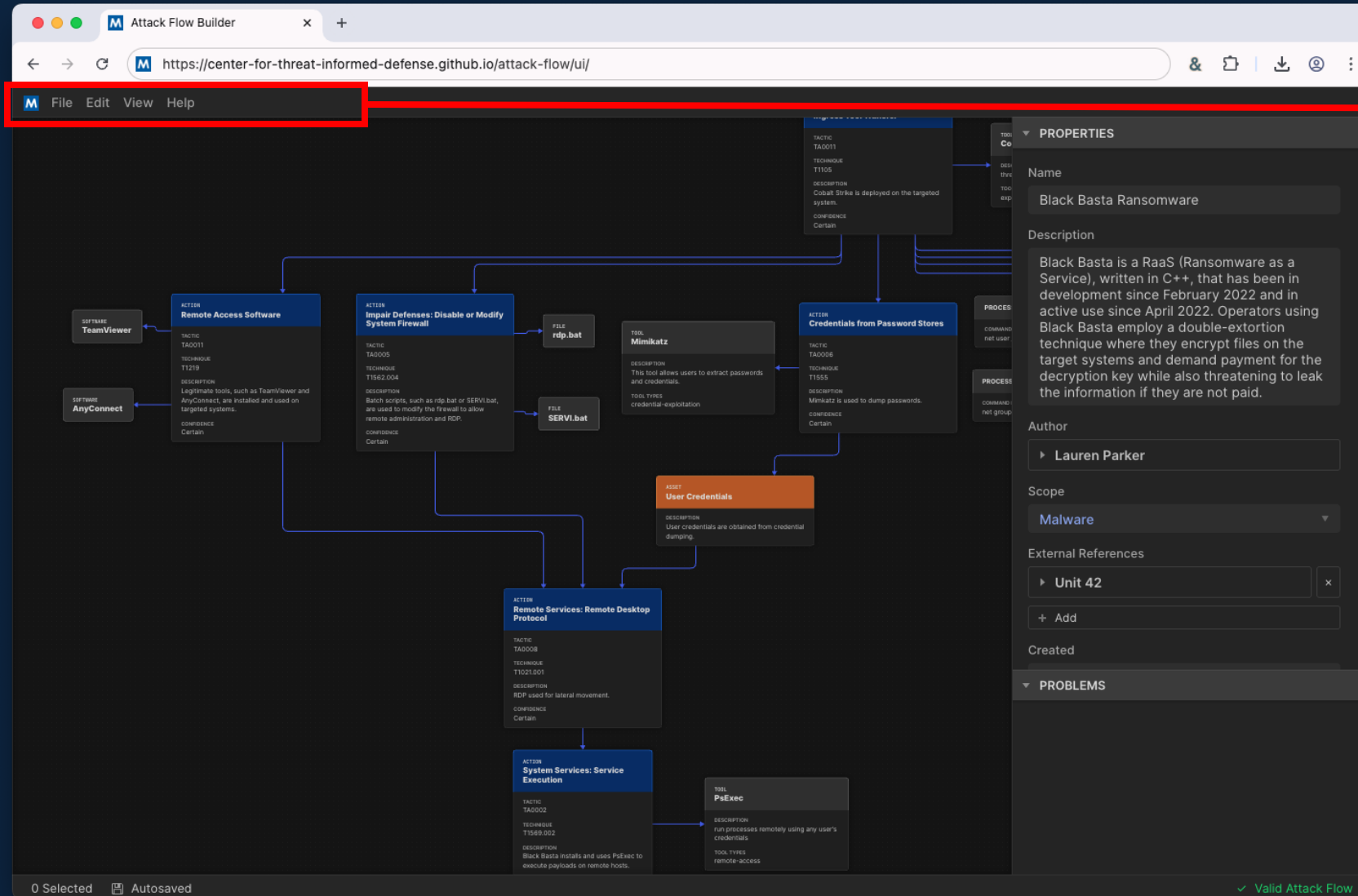
# Overview of Attack Flow Builder

# Web App for Diagramming Attacks

- Open source, web-based tool.
- Similar to Visio: create nodes (boxes) and connect with edges (lines).
- Create, edit, export, and present flows.
- Private: flow data stays in the browser. We do not collect or share it.
- Can be hosted at your organization for additional privacy & assurance.



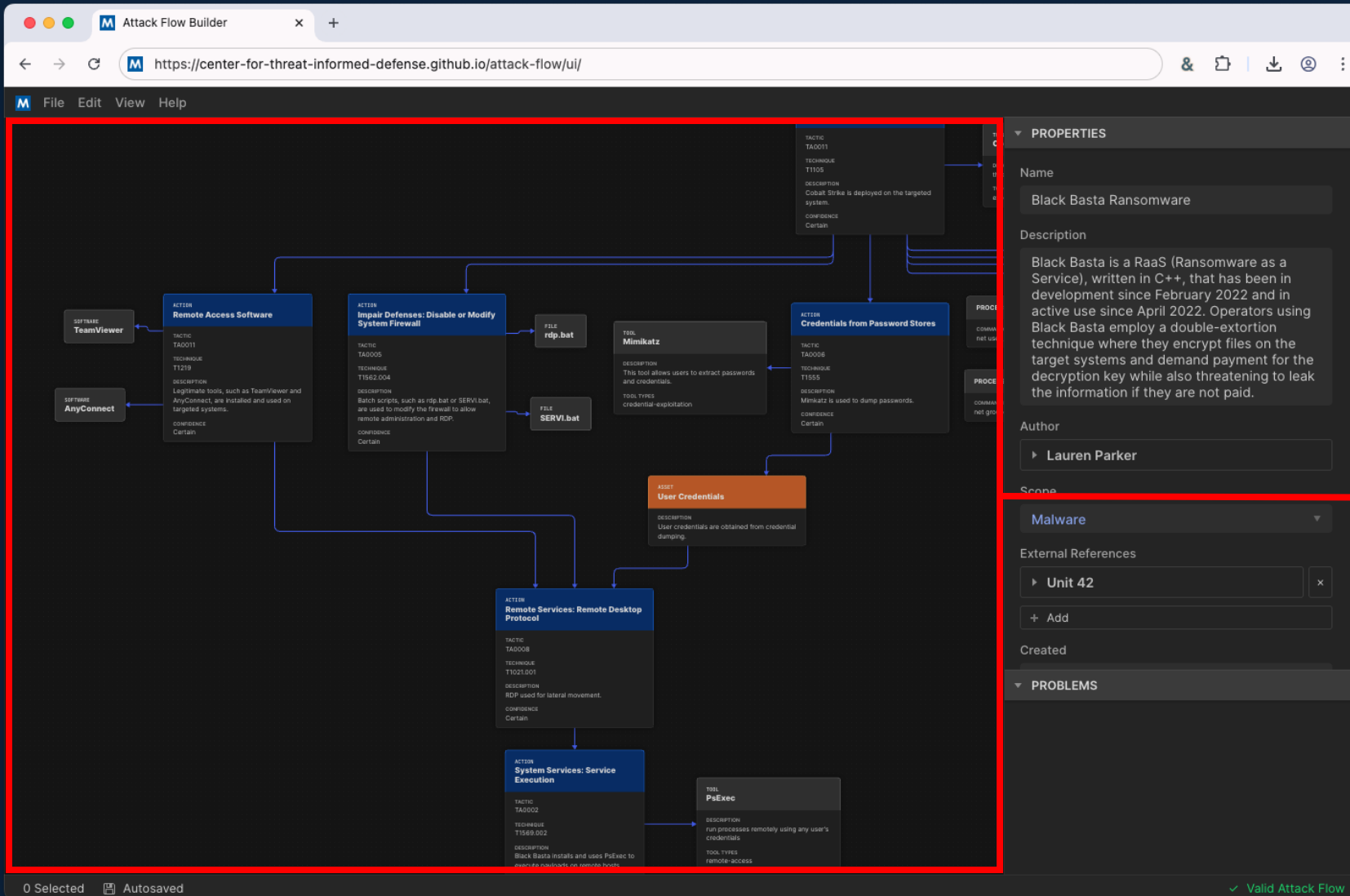
# Web App for Diagramming Attacks



Menu bar: new flow, open flow, export, copy/paste, etc.

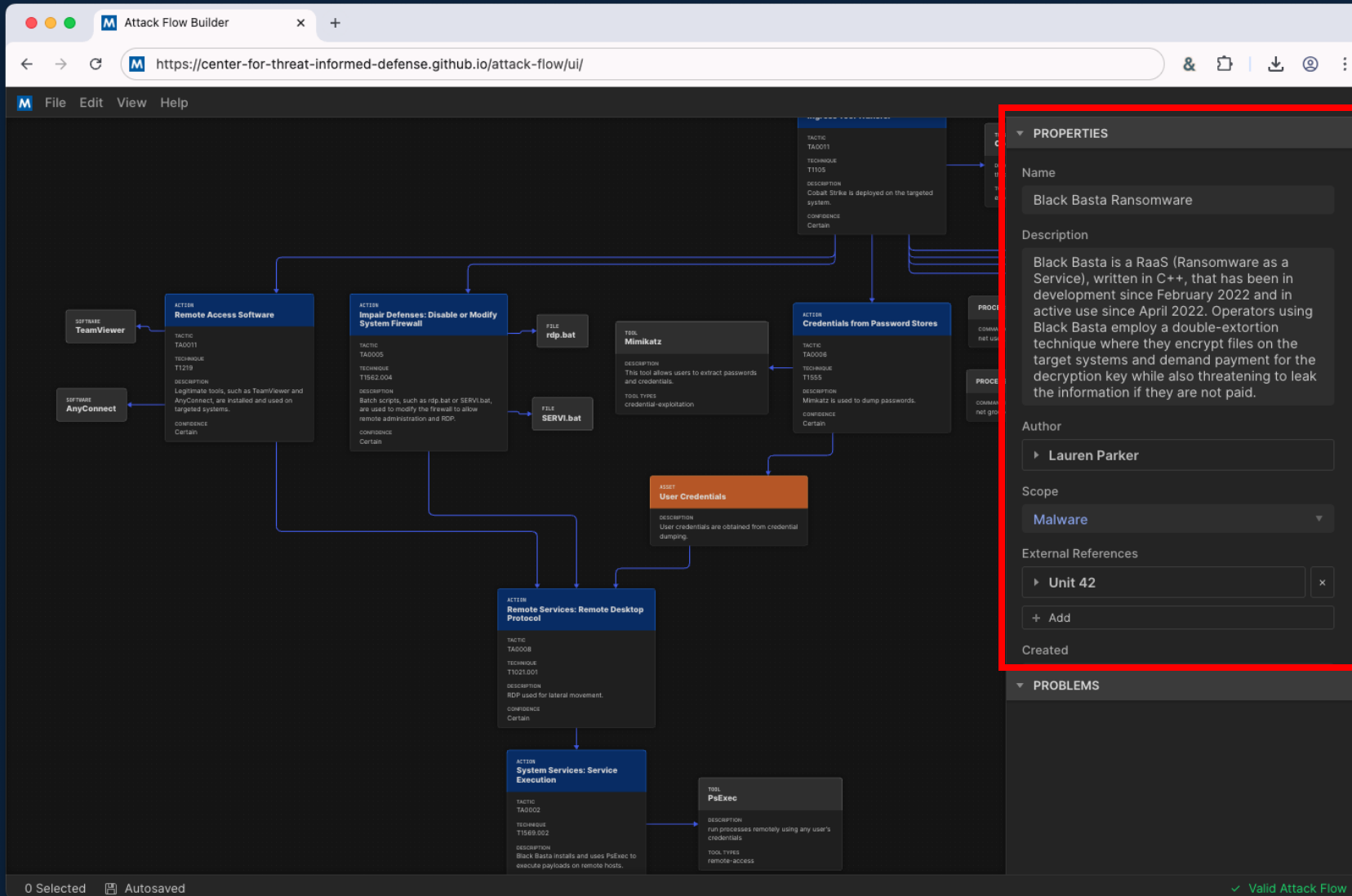
Select: Menu bar → Edit → Create → ... to get started with adding items

# Web App for Diagramming Attacks



Canvas: this is where you draw the diagram, the nodes, and the edges.

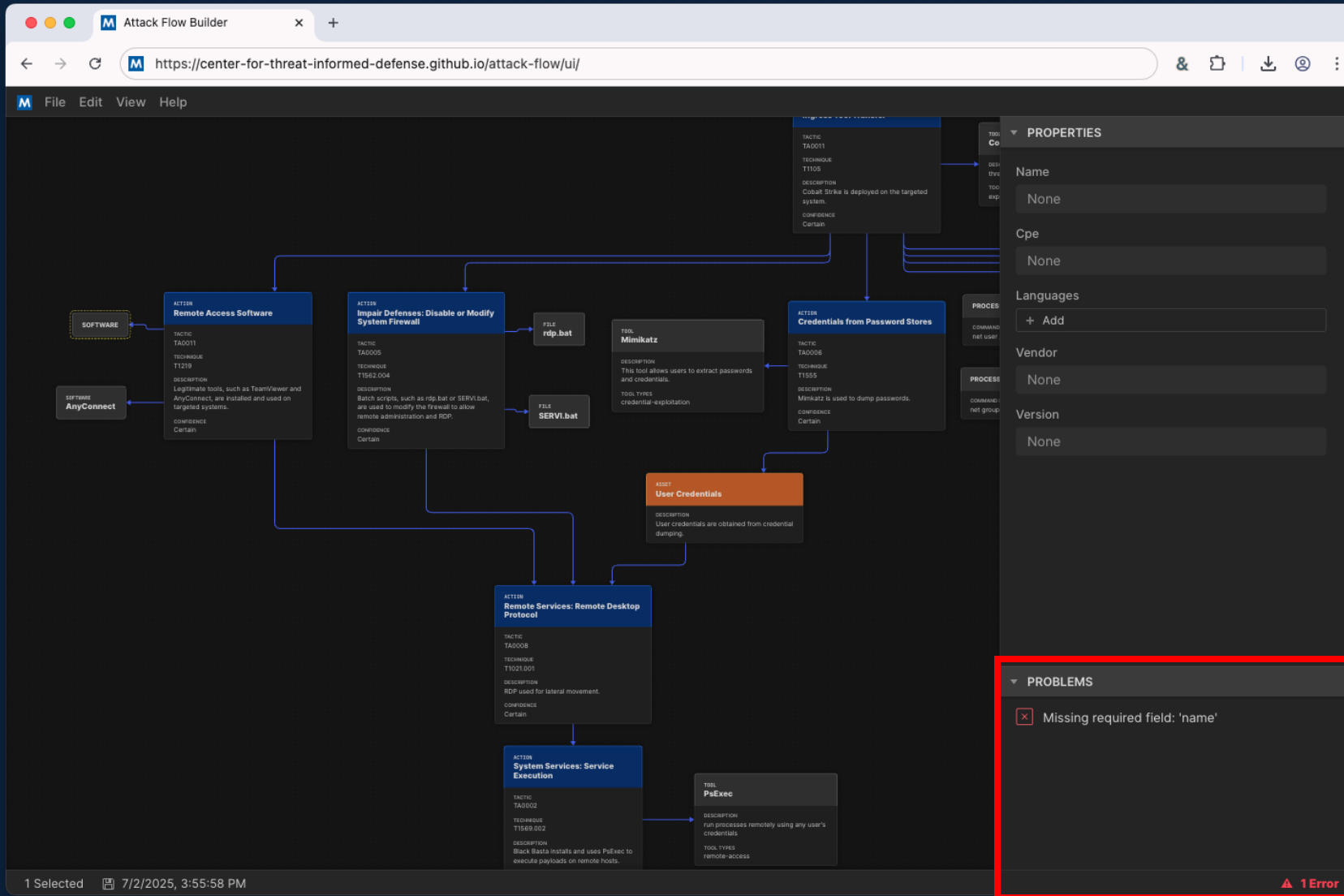
# Web App for Diagramming Attacks



Properties: when you select an item in the diagram, its properties appear here and you may edit them



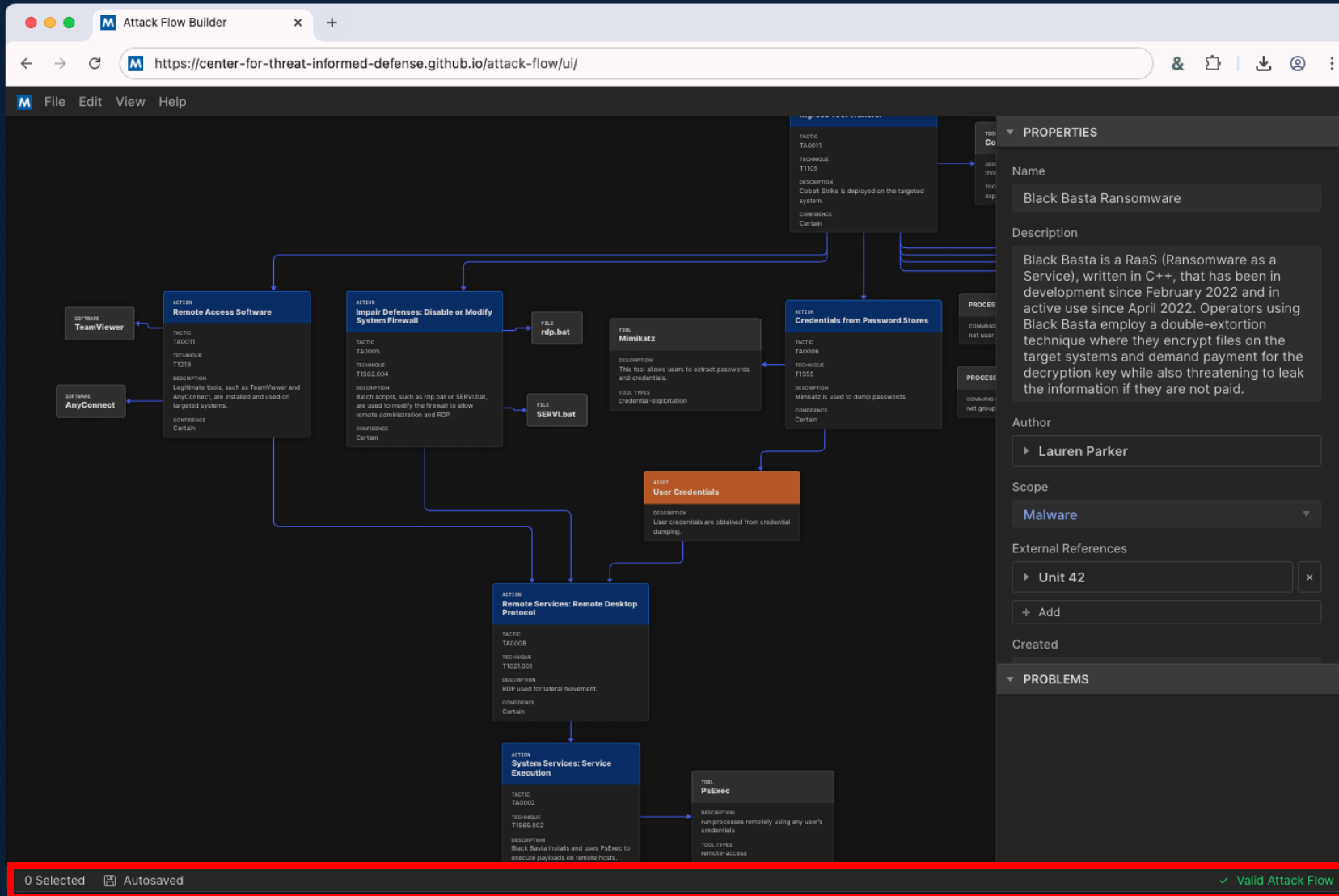
# Web App for Diagramming Attacks



Validation: displays any problems with the flow. click on an issue to zoom to the affected item



# Web App for Diagramming Attacks



→ Footer

# Attack Flow Building Blocks

# Building Blocks: Action

<b>ACTION</b> <b>Phishing: Spearphishing Attachment</b>
<b>TACTIC</b> [ENT] TA0001 Initial Access
<b>TECHNIQUE</b> [ENT] T1566.001 Spearphishing Attachment
<b>DESCRIPTION</b> Victims receive spear phishing emails with malicious zip files attached.
<b>CONFIDENCE</b> Certain

- Actions are the backbone of Attack Flow. They describe what the adversary is doing at the TTP level (tactic, technique, procedure).
- An action should at least have a name and description. The name is displayed in the blue header, the description is displayed underneath.
- Each action may be mapped to ATT&CK but not required to do so. (It is called “*Attack Flow*”, not “ATT&CK Flow”.)

# Building Blocks: Action Properties

**ACTION**

**Phishing: Spearphishing Attachment**

**TACTIC**

[ENT] TA0001 Initial Access

**TECHNIQUE**

[ENT] T1566.001 Spearphishing Attachment

**DESCRIPTION**

Victims receive spear phishing emails with malicious zip files attached.

**CONFIDENCE**

Certain



Click on canvas to display in sidebar

▼ PROPERTIES

Name

Phishing: Spearphishing Attachment

TTP Mapping

TACT.	TA0001
TECH.	T1566.001

Description

Victims receive spear phishing emails with malicious zip files attached.

Confidence

Certain ▼

Execution Start

🕒	None
🌐	America/New York ▼

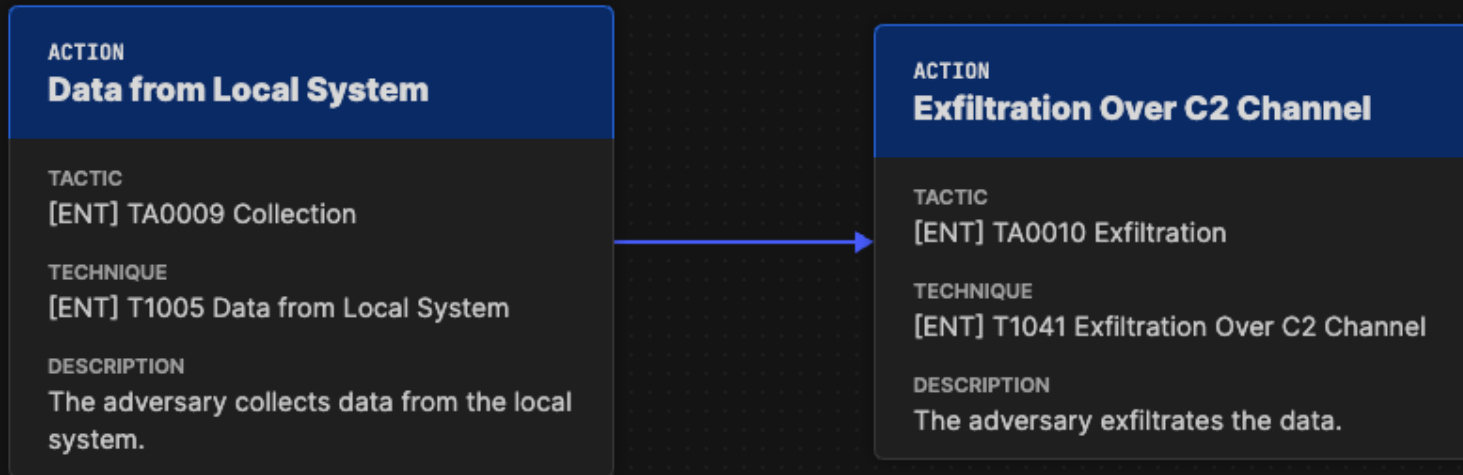
Execution End

🕒	None
🌐	America/New York ▼

# Building Blocks: Action Confidence

Term	Description	Confidence Value	Confidence Range
Speculation	Information that is purely speculative or hypothetical, e.g. the author imagines a what-if scenario.	0	0-0
Very Doubtful	Information that is very unlikely to be true. All of the available evidence is against it, or it may have bias in its reporting, e.g. an adversary providing attribution information.	10	1-20
Doubtful	Information that is unlikely to be true. Most of the available evidence is against it.	30	21-40
Even Odds	Information that is equally like to be true as not true; a coin flip. The available evidence is equally weighted in support and against.	50	41-60
Probable	Information that is likely to be true. Most of the available evidence supports it.	70	61-80
Very Probable	Information that is very likely to be true. All of the available evidence supports it.	90	81-99
Certainty	Information that is unquestionably true.	100	100-100

# Building Blocks: Action Connections

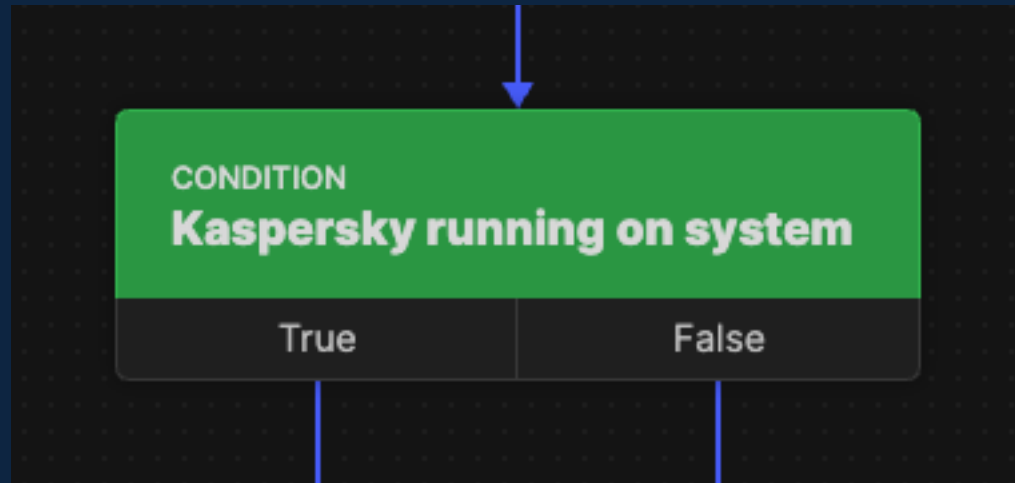


When an action is connected to another action, it represents a precondition that must be satisfied.

The second action cannot occur unless the first action happens.

**The arrows are not chronological!**

# Building Blocks: Condition

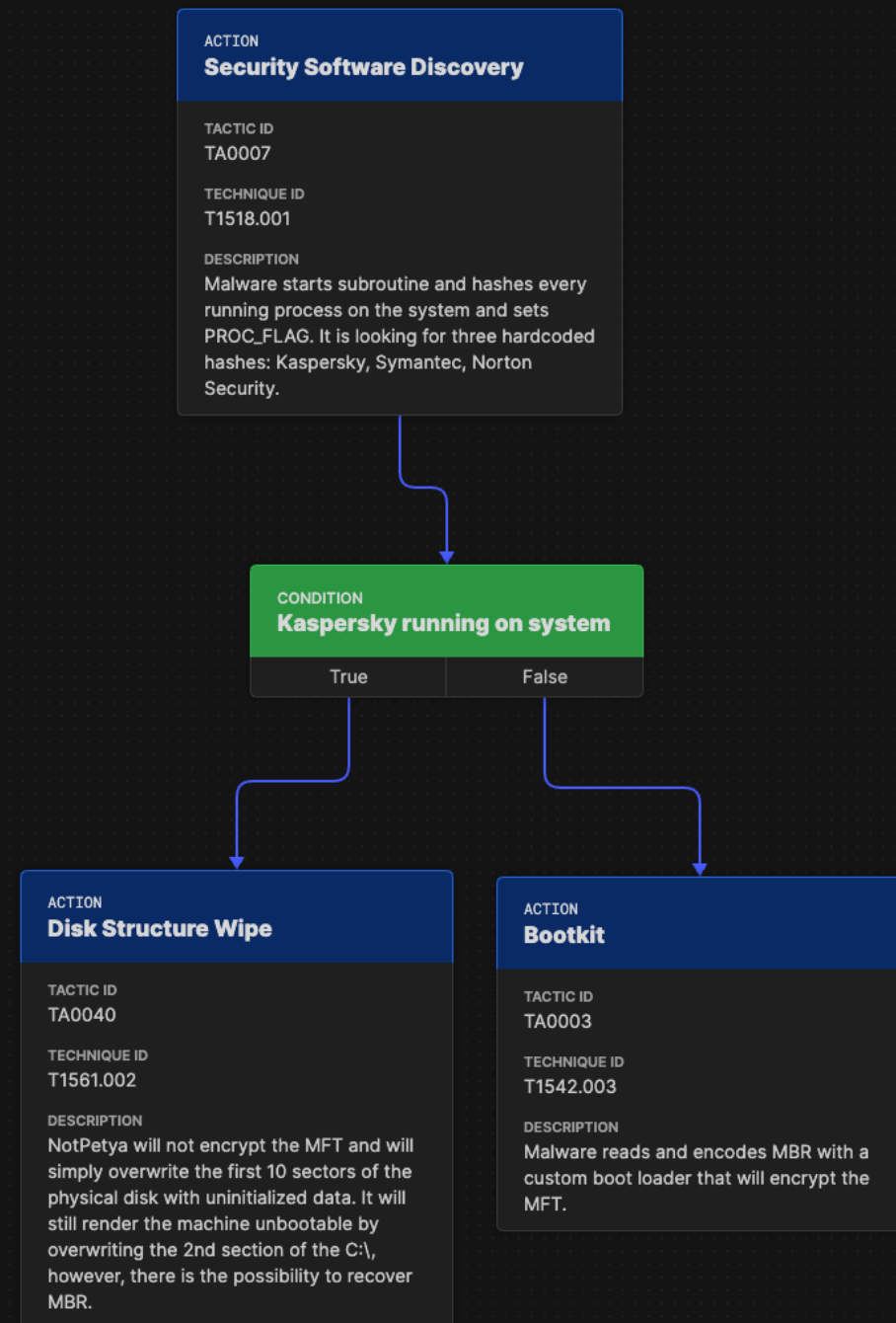


- Conditions are used to model decision points in the attack flow, show how the adversary responds to failed attempts, or to represent the state of an asset.
- The description is a human-readable text that is displayed in the green header.
- Can optionally use the STIX Pattern language for machine-readable condition evaluation.
- Unlike other nodes, conditions have two types of ports (true, false).



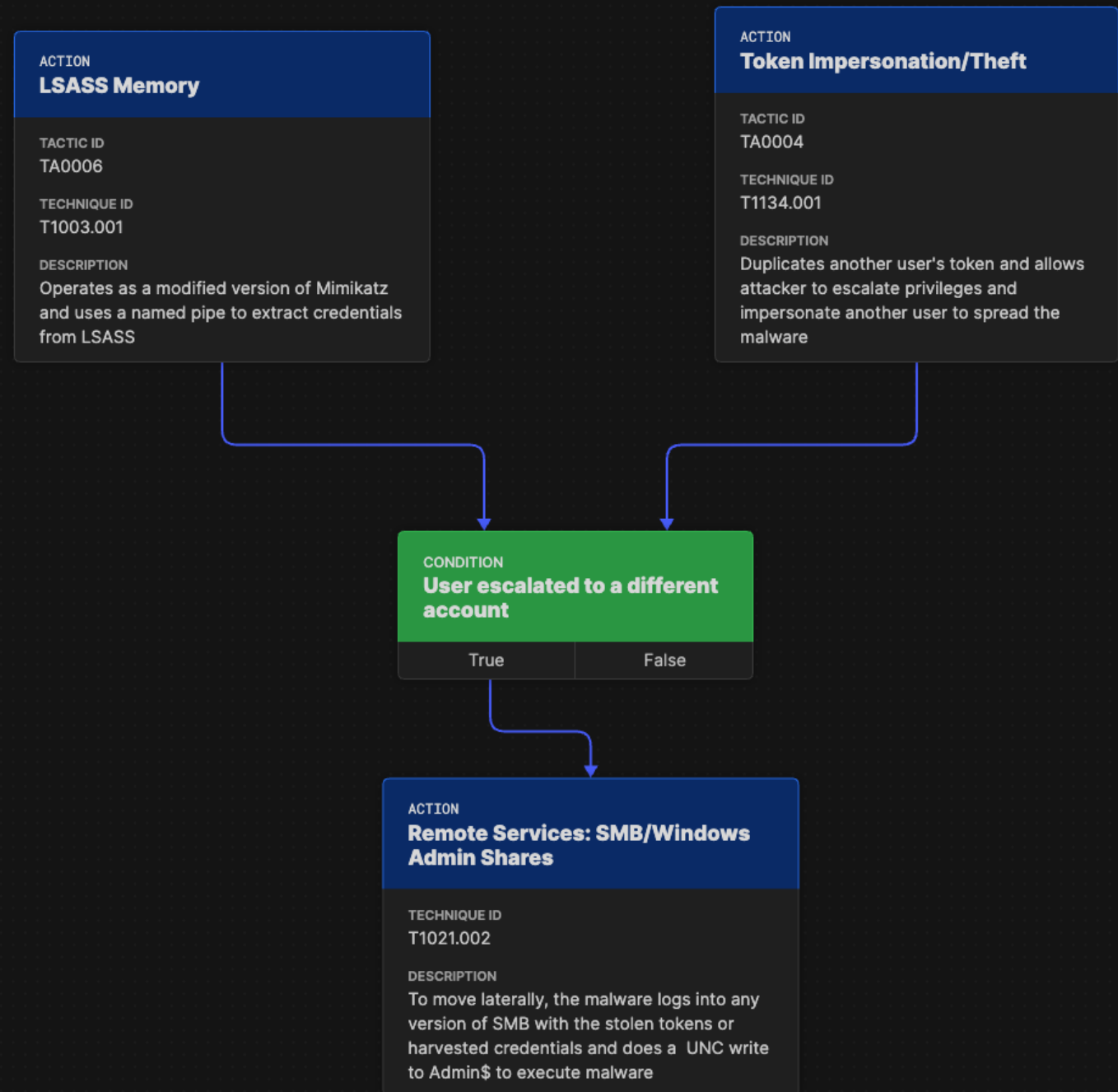
# Building Blocks: Condition

- NotPetya checks if Kaspersky is running on the system.
  - If yes: corrupts the Master Boot Record.
  - If no: installs a malicious bootloader.

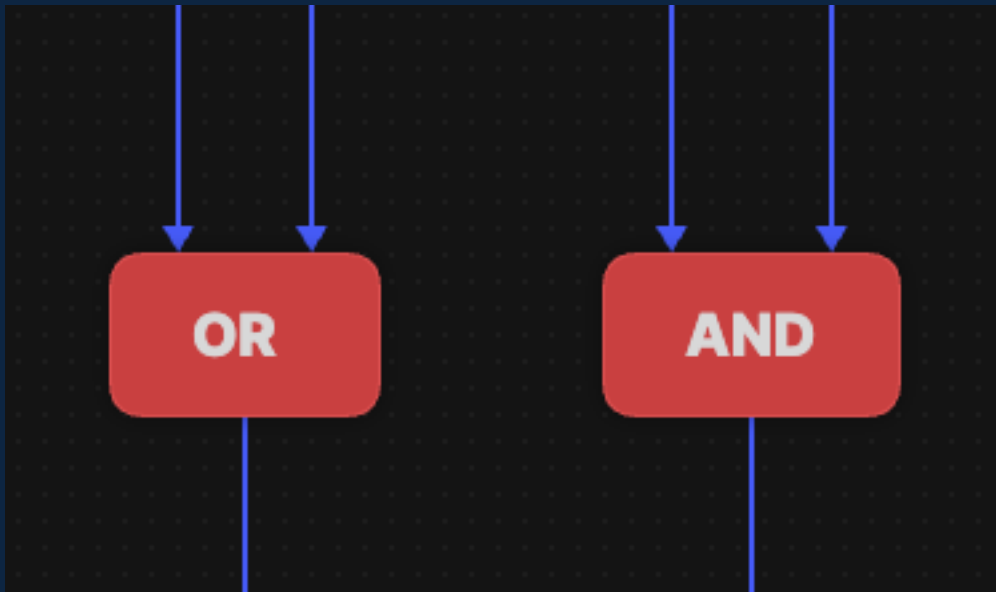


# Building Blocks: Condition

- Helpful for easing understanding of complex topics.
- E.g. in NotPetya, the privilege elevation is complex and requires knowledge of Windows internals.
- The condition object makes the flow easier to read.



# Building Blocks: Operator



- Operators allow multiple attack paths to converge.
- The OR operator requires *any* of its inputs to succeed before execution continues.
- The AND operator requires *all* its inputs to succeed before execution continues.

# Building Blocks: OR Operator

- NotPetya has two privilege escalation techniques.
- If *either one* succeeds, then it it can execute its lateral movement technique.

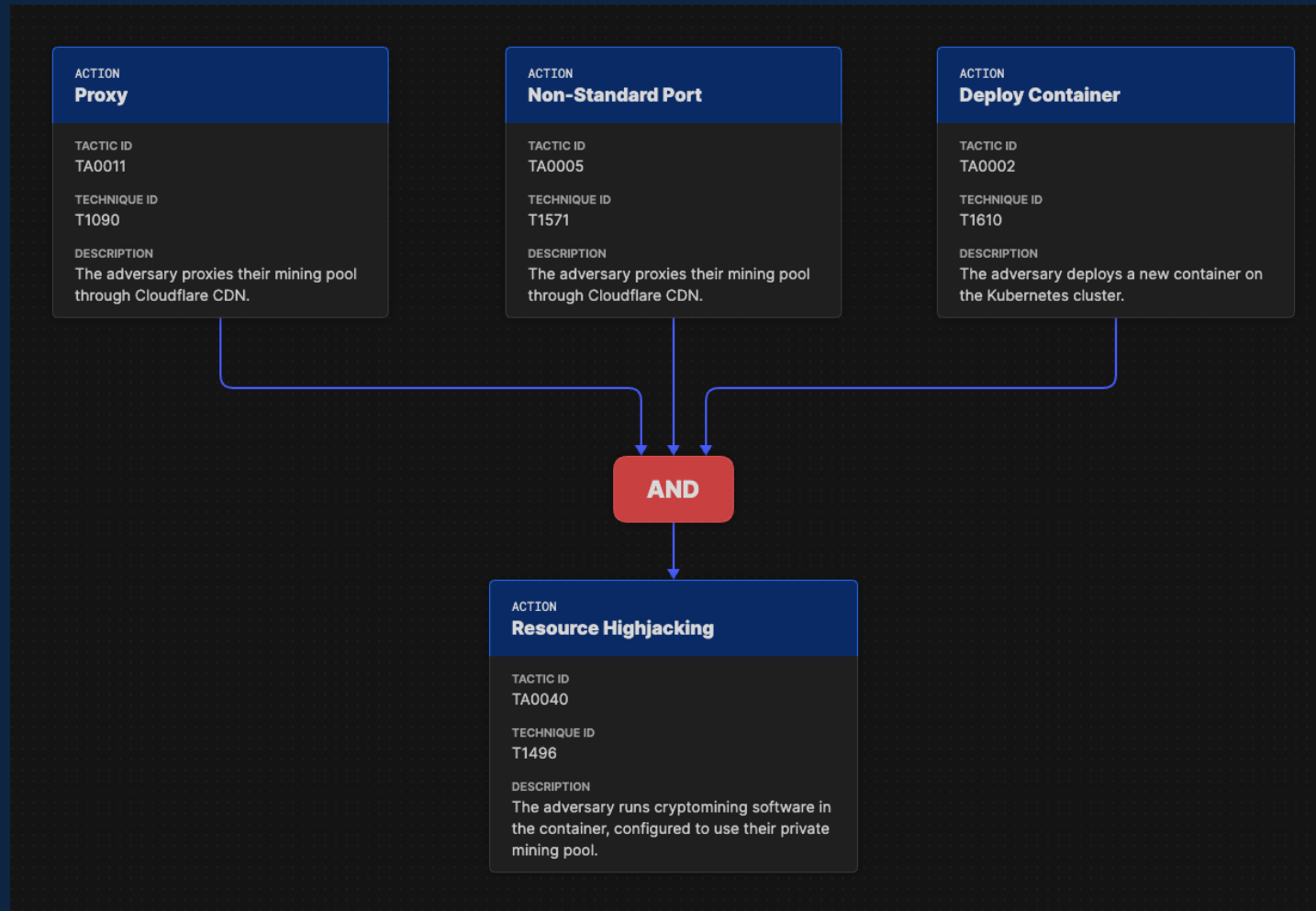
ACTION
<b>LSASS Memory</b>
TACTIC ID TA0006
TECHNIQUE ID T1003.001
DESCRIPTION Operates as a modified version of Mimikatz and uses a named pipe to extract credentials from LSASS

ACTION
<b>Token Impersonation/Theft</b>
TACTIC ID TA0004
TECHNIQUE ID T1134.001
DESCRIPTION Duplicates another user's token and allows attacker to escalate privileges and impersonate another user to spread the malware

OR

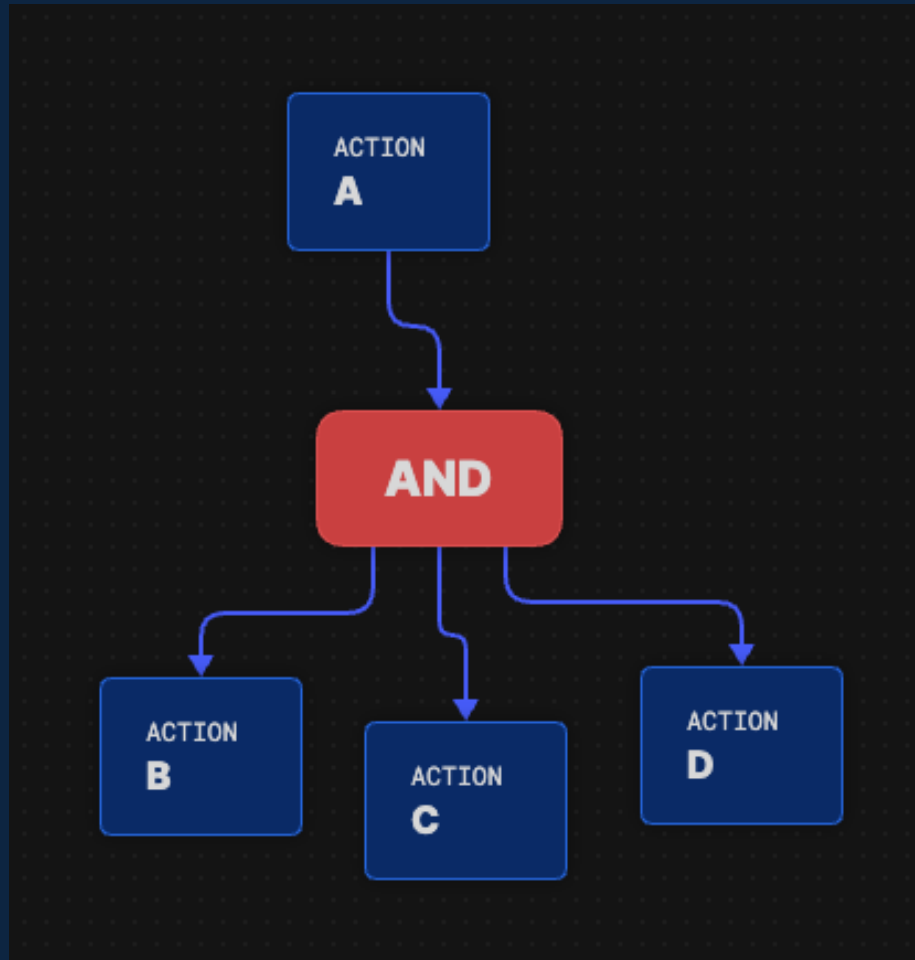
ACTION
<b>Remote Services: SMB/Windows Admin Shares</b>
TECHNIQUE ID T1021.002
DESCRIPTION To move laterally, the malware logs into any version of SMB with the stolen tokens or harvested credentials and does a UNC write to Admin\$ to execute malware

# Building Blocks: AND Operator



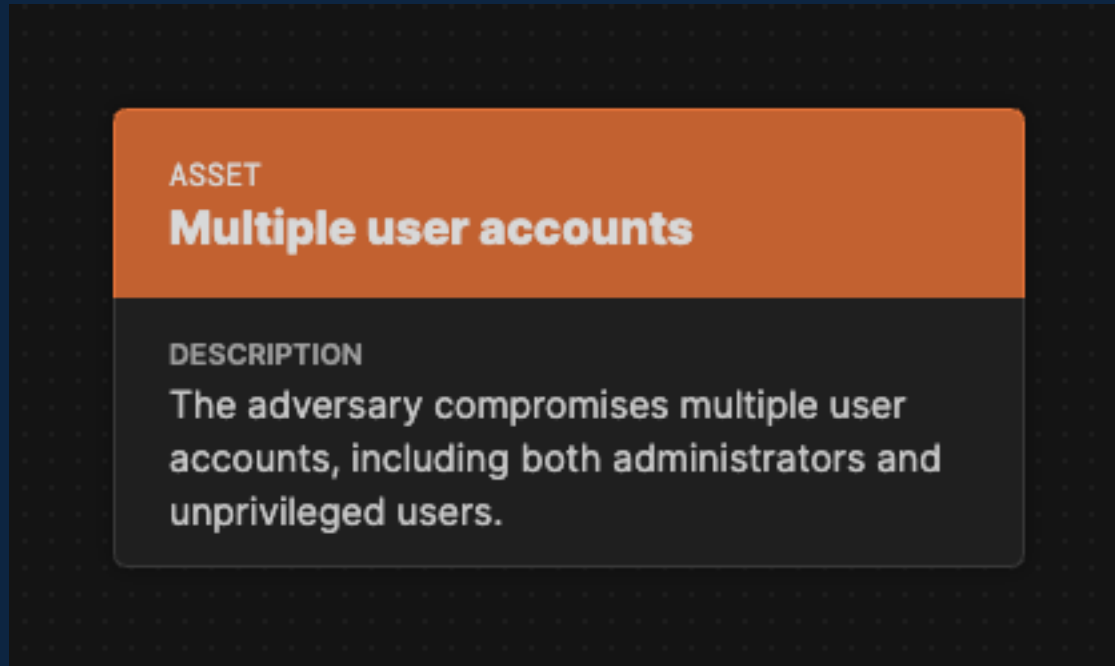
- In the Tesla flow, the adversary must position infrastructure, configure a non-standard port, and deploy a Kubernetes container.
- If they succeed in all three, then they can execute resource hijacking.

# Building Blocks: Operator Anti-pattern



- An operator with a single input doesn't do anything.
- *This is not recommended.*
- We see this in some flows due to a misunderstanding about how to use operators.

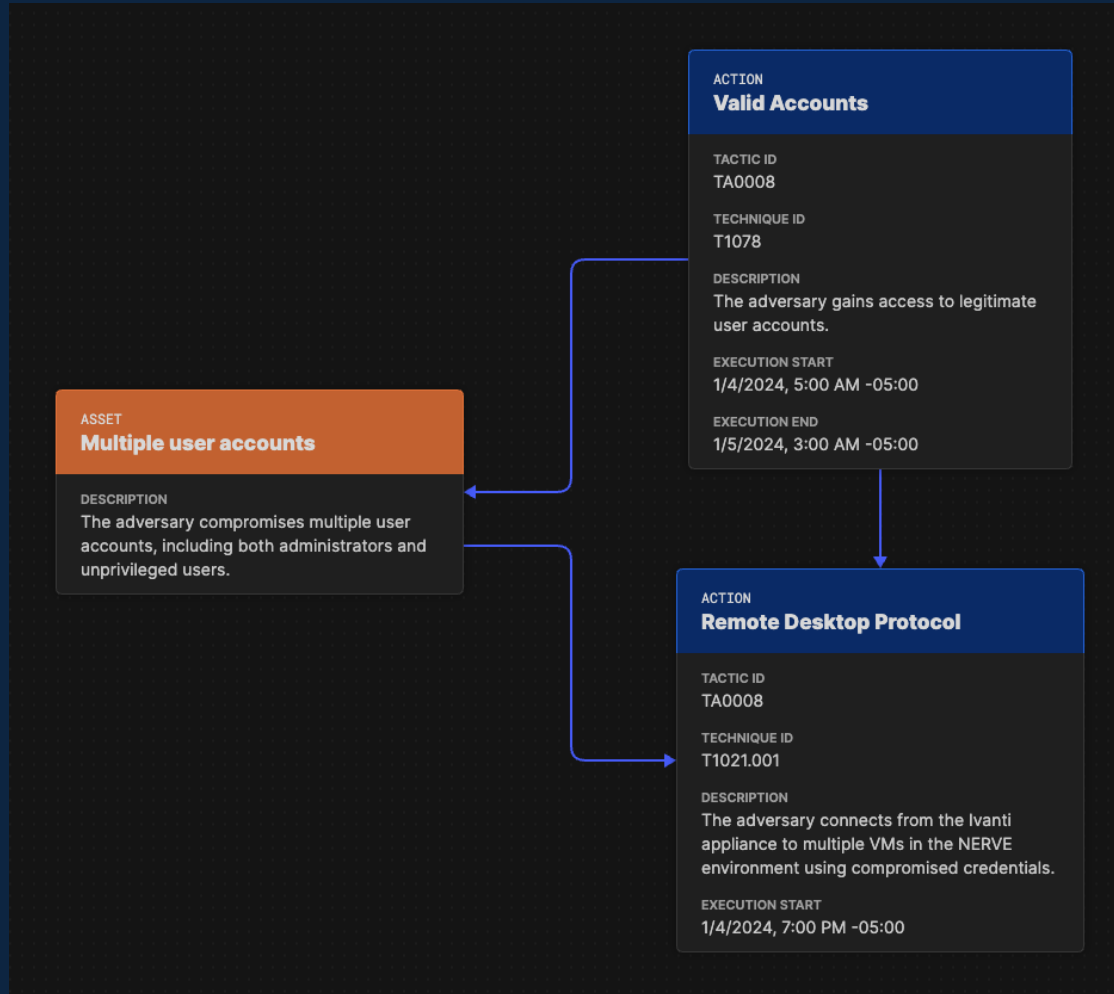
# Building Blocks: Assets



- Assets represent information systems, data, or users involved in an attack.
- Very generalized: it contains only a name and a description.
- You can add additional structured data using STIX.



# Building Blocks: Asset Connections



- An edge from an action to an asset indicates that the action modifies the state of that asset.
- An edge from an asset to an action indicates that the action depends on the state of that asset.

# Building Blocks: Asset Anti-pattern

- Actions should connect to other actions, conditions, or operators in a continuous chain.
- No other nodes should come between actions.

**ACTION**  
**Valid Accounts**

TACTIC ID  
TA0008

TECHNIQUE ID  
T1078

DESCRIPTION  
The adversary gains access to legitimate user accounts.

EXECUTION START  
1/4/2024, 5:00 AM -05:00

EXECUTION END  
1/5/2024, 3:00 AM -05:00



**ASSET**  
**Multiple user accounts**

DESCRIPTION  
The adversary compromises multiple user accounts, including both administrators and unprivileged users.



**ACTION**  
**Remote Desktop Protocol**

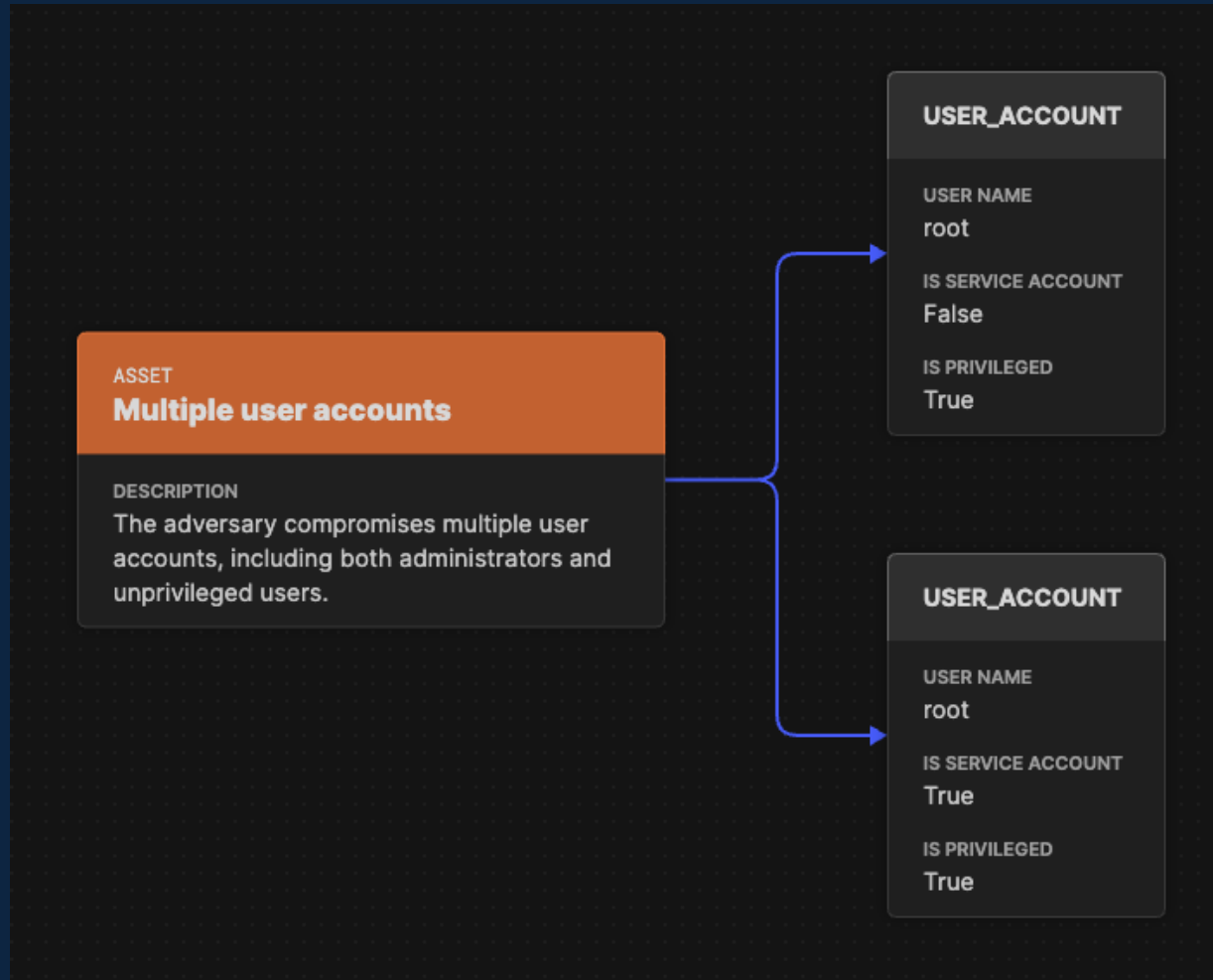
TACTIC ID  
TA0008

TECHNIQUE ID  
T1021.001

DESCRIPTION  
The adversary connects from the Ivanti appliance to multiple VMs in the NERVE environment using compromised credentials.

EXECUTION START  
1/4/2024, 7:00 PM -05:00

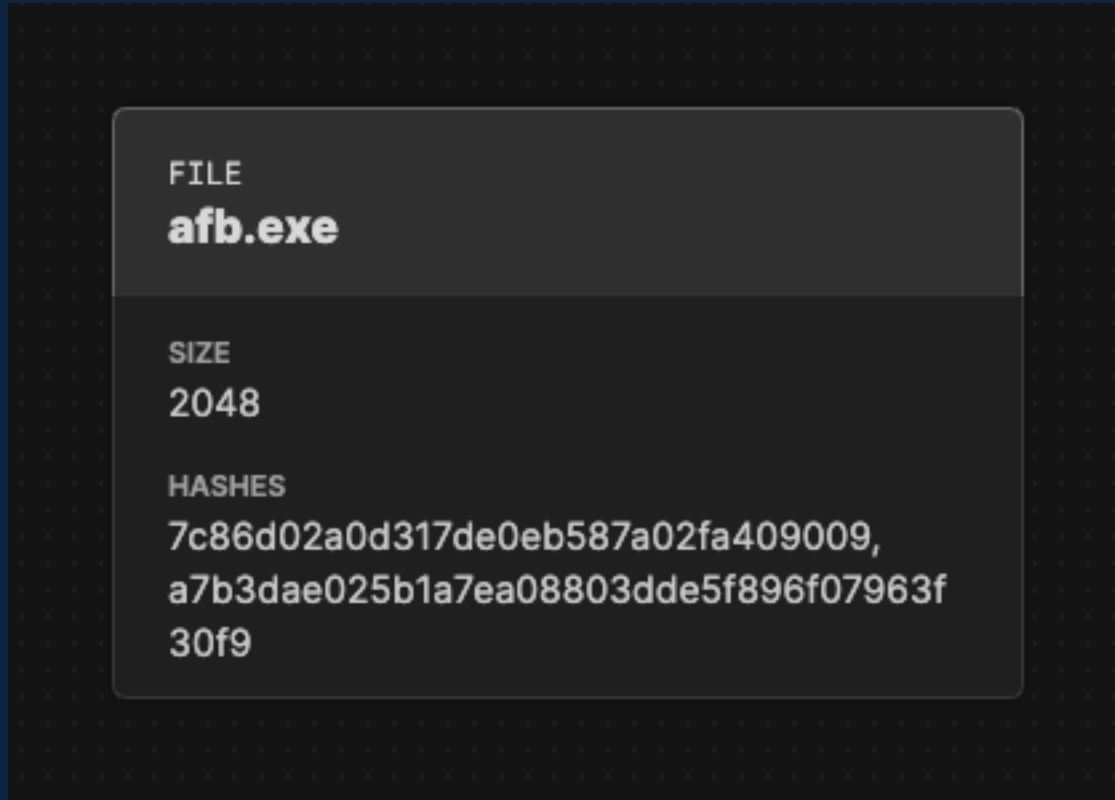
# Building Blocks: Asset Structured Data



- While assets are very simple on their own (just name and description) they can be enriched using additional nodes.
- For example, add “User Account” nodes to an asset; this creates structured data about the impacted accounts.

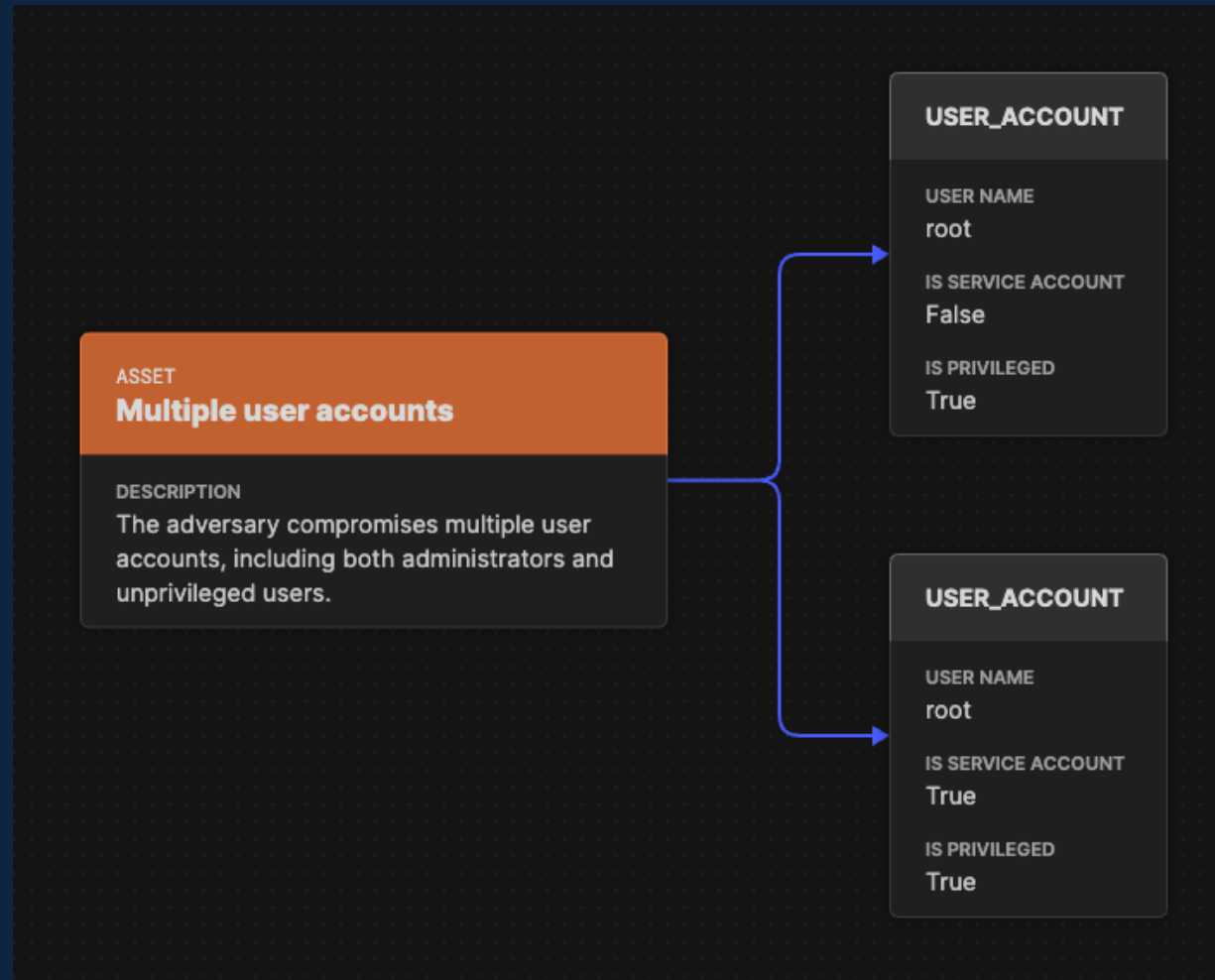
# Attack Flow STIX Nodes

# STIX Nodes



- Attack flow supports all STIX 2.0 types:
  - 18 STIX Domain Object (SDO) Types
  - 18 STIX Cyber Observable (SCO) Types
- These types are defined in the STIX specification; we adhere to that.
- STIX nodes can be connected with edges just like any other node.
- You can export an Attack Flow to a STIX bundle and import it into any tool that can process STIX (e.g., OpenCTI, etc.)

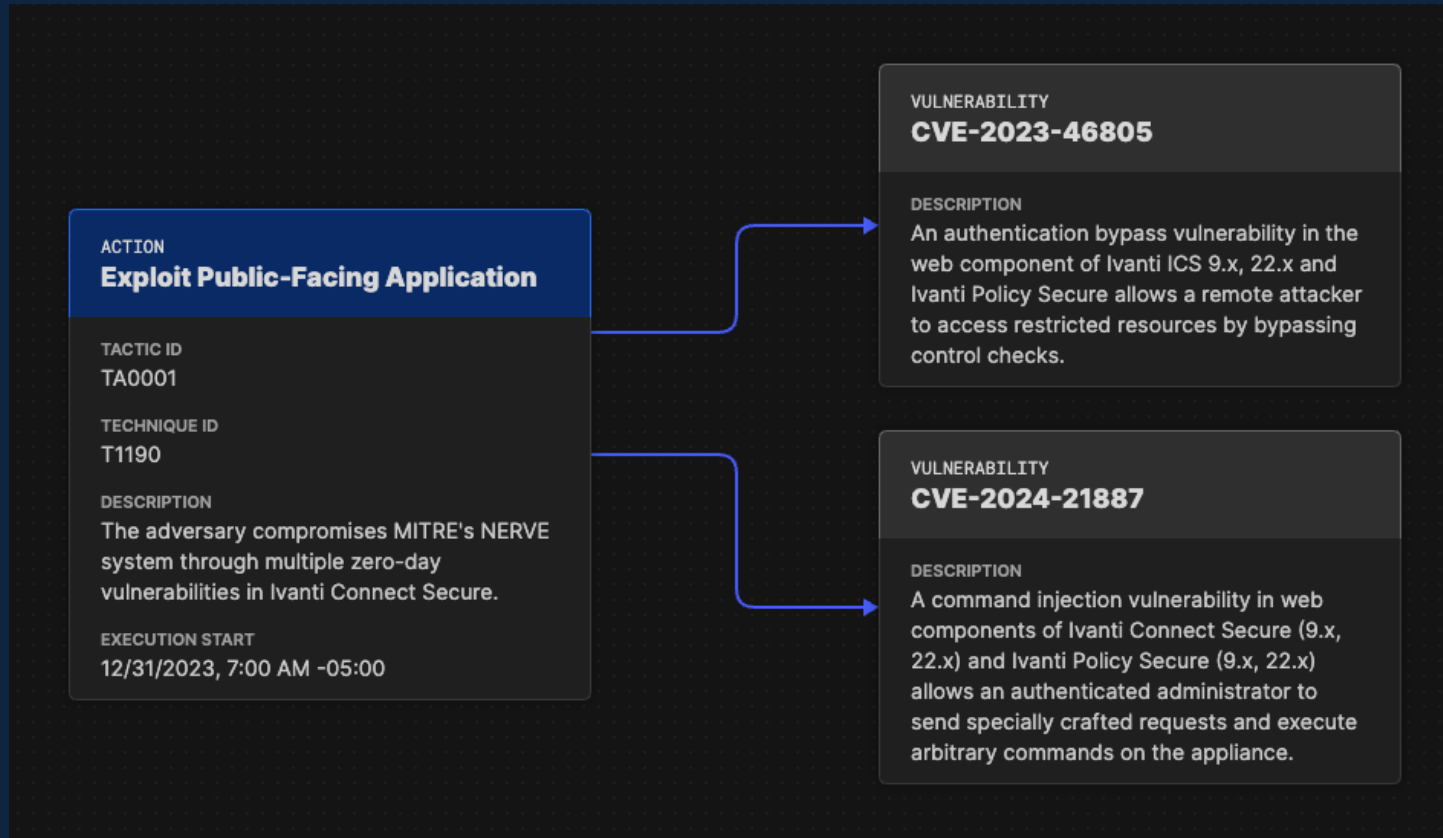
# STIX Nodes



This example (that we saw a few slides ago) shows the “User Account” STIX object enriching an Asset node.

In STIX this is represented by the "user-account--" observable type.

# STIX Nodes



This example shows the "Vulnerability" STIX type used to enrich an Action node.



# STIX Nodes

- All of the properties defined in STIX are settable and viewable in Attack Flow Builder.
- For example, “File” is a STIX Observable with 9 property fields.

The screenshot displays the Attack Flow Builder interface. On the left, a 'FILE' node is shown with the name 'afb.exe'. It lists the following properties: SIZE (2048) and HASHES (7c86d02a0d317de0eb587a02fa409009, a7b3dae025b1a7ea08803dde5f896f07963f30f9). On the right, the 'PROPERTIES' panel is expanded, showing the same fields for configuration. The 'Name' field is set to 'afb.exe', 'Name Enc' is 'None', 'Size' is '2048', and 'Magic Number Hex' is 'None'. The 'Hashes' section contains two entries with expand/collapse and delete icons, and an 'Add' button.

PROPERTY	VALUE
FILE	afb.exe
SIZE	2048
HASHES	7c86d02a0d317de0eb587a02fa409009, a7b3dae025b1a7ea08803dde5f896f07963f30f9

**PROPERTIES**

Name: afb.exe

Name Enc: None

Size: 2048

Hashes:

- 7c86d02a0d317de0eb587a02fa... [x]
- a7b3dae025b1a7ea08803dde5f... [x]
- + Add

Magic Number Hex: None

# STIX Data Validation

STIX is easy to work with in Attack Flow.

The validator verifies correct formatting.

The screenshot displays the STIX Data Validation interface within the Attack Flow tool. It features a central panel showing the details of a file object named **afb.exe**. The file's size is listed as 2048 bytes, and it has two associated hashes: **AAAA** (MD5) and **a7b3dae025b1a7ea08803dde5f896f07963f30f9**. The **PROBLEMS** section on the right indicates an **Invalid hash value.** error. A red banner at the bottom of the interface states **Invalid Attack Flow**.

PROPERTY	VALUE
Name	afb.exe
Name Enc	None
Size	2048
Hashes	AAAA, a7b3dae025b1a7ea08803dde5f896f07963f30f9
Hash Type	MD5
Hash Value	AAAA

**PROBLEMS**

- Invalid hash value.

**Invalid Attack Flow**

# Demo

# End of Section 2