MITRE

MTR240221
MITRE TECHNICAL REPORT

Threat Modeling with ATT&CK

Sponsor: Center for Threat Informed Defense
Dept. No.: A400
Project No.: 840002.01.002.1083.AAA

McLean, VA

UNCLASSIFIED

How to model threats against a system using industry standard threat modeling methodologies and the ATT&CK knowledgebase

Authors:
Tyler Schechter
Ben Ballard
Dr. Kyle Wallace
Courtney Hassenfeldt
Tiffany Bergeron

July 2024

Executive Summary

The process outlined in this paper details an approach developed by MITRE Engenuity's Center for Threat-Informed Defense (hereafter, the Center) for integrating MITRE ATT&CK® into your organization's existing threat modeling methodology. **The latest version of our work, along with video tutorials and more, can be found on our website which will be accessible through the Center's site July 25th, 2024.** At the core of this approach are four key questions, outlined in the Threat Modeling Manifesto[4], that we need to answer:

- Question 1 – What are we working on?
- Question 2 – What could go wrong?
- Question 3 – What are we going to do about it?
- Question 4 – Did we do a good job?

This process is intended for universal application to any system or technology stack (large or small) using any existing threat modeling methodology like STRIDE, PASTA, or Attack Trees. To demonstrate its use and applicability to a wide audience of cybersecurity practitioners, we apply this process to a fictional internet of things (IOT) system called the Ankle Monitoring Predictor of Stroke (AMPS). The fictional AMPS device gives the wearer and their healthcare providers indications and warnings of a stroke. The systems and subsystems that make up this device are modeled after a popular commercially available IOT device and intentionally chosen for their mobile/cloud-based dependencies. This broad application to a system spanning mobile and enterprise environments allows readers to visualize how this process could be applied to their problem sets. Examples throughout this paper are from the perspective of a security team working for the AMPS manufacturer. They have been tasked with modeling threats to the AMPS device and supporting system infrastructure.

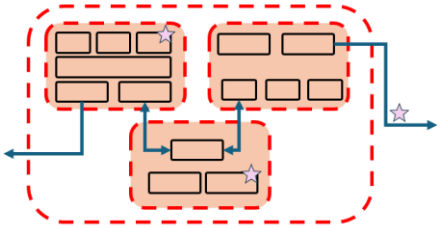
Using the process described throughout this paper, we identify critical components of the AMPS, prioritize threats to those components, and recommend mitigations. Threat modeling with ATT&CK allows us to leverage data from the Cyber Threat Intelligence (CTI) community and significantly improve our results in Questions 2 and 3. The below graphic is an overview of our recommended process to answer these questions. We will break down our means of answering each question in further detail throughout the paper.

[4] https://www.threatmodelingmanifesto.org/


THREAT MODELING
QUESTION 1
What are we working on?
QUESTION 2
What could go wrong?
QUESTION 3
What are we going to do about it?
QUESTION 4
Did we do a good job?
Mission Decomposition
System Decomposition
Vulnerability Identification
Cyber Threat Intelligence
Defense & Risk Analysis
Mitigation & Remediation
Monitoring Analysis & Evaluation
With ATT&CK®

Condensed Process

What are we working on?



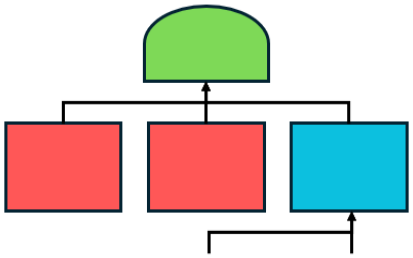
1. Develop a top-level Dataflow Diagram for your system

2. Identify critical components and dataflows that, when impacted, would result in mission failure

What could go wrong?

3. Analyze your DFD using a structured brainstorming technique (Attack Tree, STRIDE, etc.)

4. Brainstorm ATT&CK TTPs that could be used to attack the critical components within your DFD



> *You can gather ideas from TTPs previously used against your tech platform – see the ATT&CK matrix and select by platform or use the Center's Top ATT&CK Techniques Calculator.*

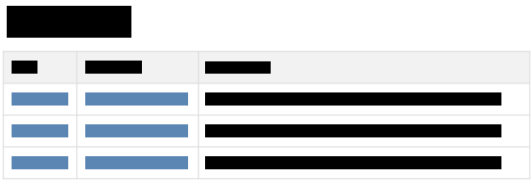
5. Once you've got your list of brainstormed TTPs, search through your existing security stack for your current ability to defend against them.

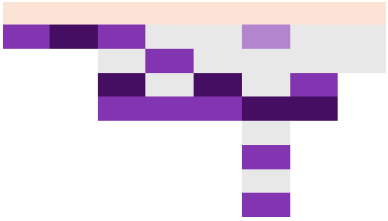
What are we going to do about it?

6. (a) Implement the mitigations listed within the ATT&CK page for each brainstormed TTP

OR

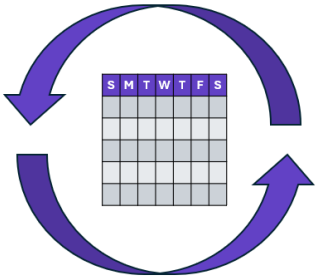
6. (b) Implement the NIST 800-53 controls for each brainstormed TTP using the MITRE Engenuity Mappings Explorer



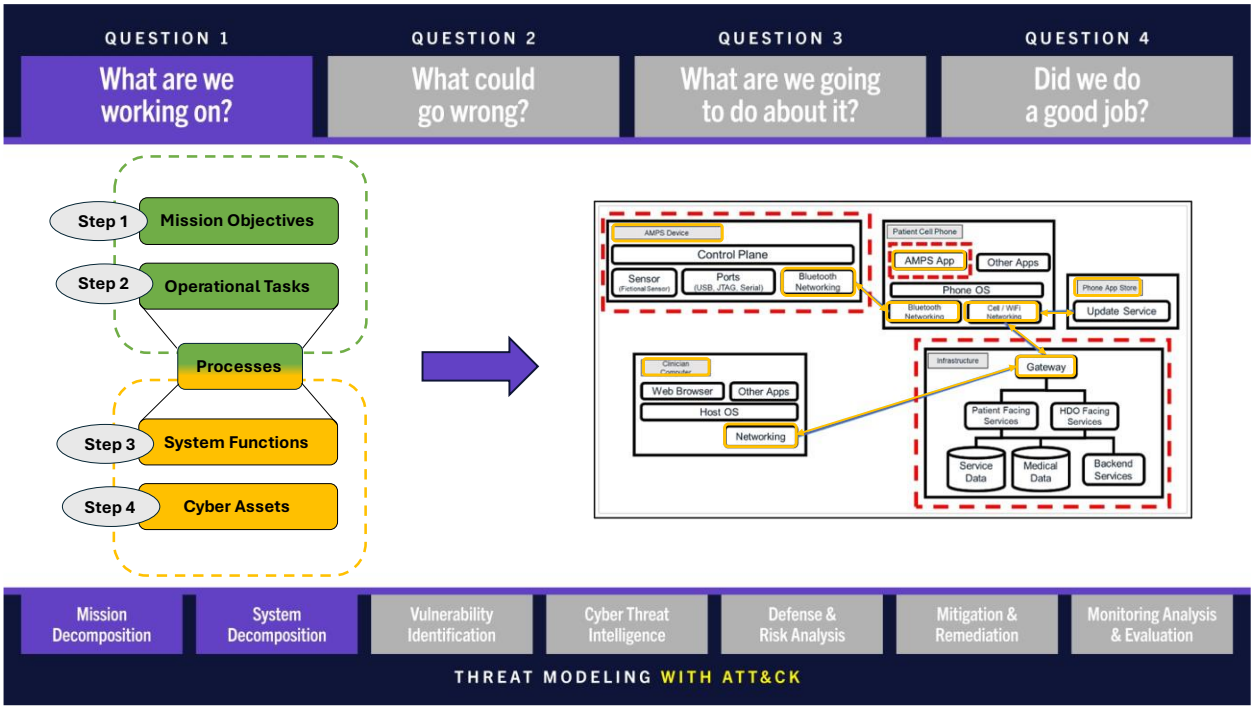


Did we do a good job?

7. Periodically repeat this process to evaluate your existing mitigations and make sure they are in sync with the development of your system.



Question 1: What are we working on?



Question 1 enables the primary and secondary function(s) of the system to be identified and analyzed. It identifies critical tasks that must be performed for the system to successfully accomplish its function(s) and highlights the resources those critical tasks rely upon.

Approach

☐ Assembling Your Team

Goal: Educate and grow your team

☐ *Step 1: Gather relevant documentation*

Ensure the team familiarizes themselves with relevant documentation of the system under analysis. This is to ensure the team understands the system prior to engaging other stakeholders. Depending on your type of system and the time you have, some documents we recommend reviewing are:

Documents of Interest	
Information security policies	Password policies
Employee staffing policies	Data classification policies
Disaster recovery plans	Data backup policies
Business continuity plans	External drive policies
Incident response plans	Network architecture
Remote access policies	Network security policy
Risk assessment procedures and policies	User permission guidelines
Allowed applications list	Account management policies
Security applications list	Cloud security policies
File storage application list & policies	Cloud architecture
Inventory procedures	Mobile security policies

☐ *Step 2: Identify key stakeholders*

These individuals are your subject matter experts who will provide insights into the nature of the assessed system. They can either serve as active members of the assessment team or as points of contact throughout the assessment as needed. You may already know who these individuals are, but there are some techniques that might help you narrow your search for relevant personnel:

1. *Pre-mortem*
    - Imagine a crisis scenario: the assessed system is inoperable; the timeline for project development has broken down; you're unable to provide a key service to your customers, etc. Who do you call? What information do you need?
2. *Responsible, Accountable, Consulted, and Informed (RACI) Matrix*
    - A RACI Matrix lets you represent which staff are involved in the operation of the system in question, and their respective impact.
    - A RACI Matrix that ties specific staff to tasks and cyber assets can be developed in tandem with the development of a Mission Impact Assessment (see below)

☐ *Step 3: Prepare for the kick-off*

Prior to your initial kick-off meeting with the individuals identified in Step 2, request precise programmatic documentation that only speaks to the materials in scope for the assessment – consider using draft documentation instead of waiting for final products.

- For external stakeholders, pairing this with a site visit would allow them to tour the host's facilities, observe a demonstration, or participate in a technical briefing.

☐ *Step 4: Meet with stakeholders*

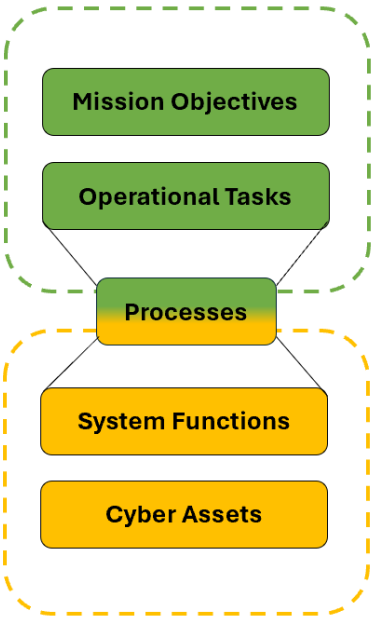
Depending on time, we recommend having at least two meetings. This gives your stakeholders a chance to communicate with each other and fosters valuable crosstalk. The first meeting is your "kick-off," which establishes a common understanding of the scope and intent of your threat modeling, establishes relationships with your identified stakeholders, and gives them an understanding of how/why you will be working with them. The second meeting is at the end of your threat modeling process and gives you a chance to present your assessments and get feedback from your stakeholders prior to completion. Informal technical exchanges between stakeholders should also occur outside of these meetings throughout the duration of the assessment.

☐ **Mission Decomposition** – Mapping system objectives (Mission Impact Analysis)

After conducting your kick-off meeting, it's time to start conducting an analysis of your mission and the systems needed to facilitate it. Drafting this analysis can also be done during your kick-off meeting or over the course of several meetings/discussions with stakeholders, depending on how much time you have. Below are four steps that can guide you through mission and system decomposition. Each step has a series of questions that drive a better understanding of your critical assets.

☐ *Step 5: Map the mission objectives*

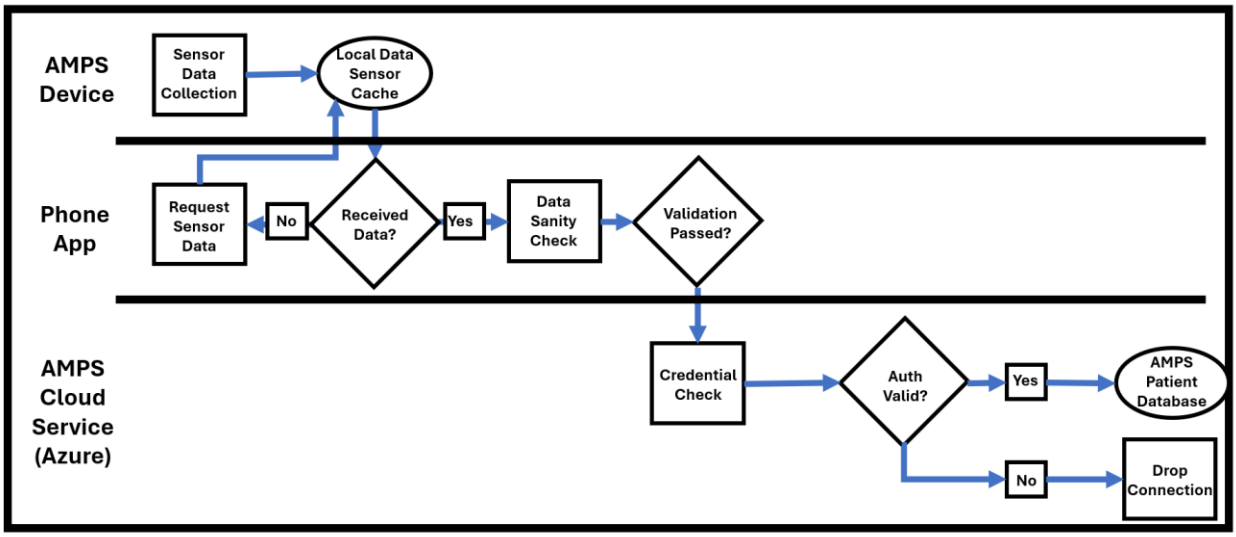
At this stage we want to determine: What is the ultimate purpose of the system? What goal is the system trying to accomplish? It's here that we'll invoke our fictional example device: the Ankle Monitoring Predictor of Stroke (AMPS). This fabricated IoT device is borrowed from MITRE's `*Playbook for Threat Modeling Medical Devices*. In our scenario, this device is meant to be worn by a patient who is at increased risk of experiencing a stroke. By wearing the device throughout the day, the patient and their doctor can monitor for indicators of an imminent stroke via a companion app on the patient's phone and readings uploaded to the AMPS cloud service each day.



As a security team evaluating AMPS for its manufacturer, we identified that a core mission objective of AMPS is to collect and share patient health data accurately and securely. Because of the sensitive nature of the health data AMPS collects and shares, which includes location data to guide an emergency response in the event of a stroke, the AMPS device should effectively protect the confidentiality of that data.

☐ *Step 6: Identify operational tasks (cross-functional flow chart)*

Next, leverage the knowledge pooled from stakeholders to determine the different operational sub-systems that contribute to the system's primary purpose identified in Step 5. An Analytic Hierarchy Process (AHP) can be used to weigh the importance of different operational systems. What are the operational tasks that must be executed to perform system's primary purpose? These tasks are also known as Mission Essential Functions (MEFs). To visualize these MEFs, we recommend using a cross-functional flow chart like the one below for the AMPS.



Image[5]: Cross-Functional Flow Chart of a Data Flow in a Fictional Medical Device: the Ankle Monitor Predictor of Stroke (AMPS)

[5] Bochniewicz, Elaine, et al. "Playbook for Threat Modeling Medical Devices." *MITRE and the Medical Device Innovation Consortium (MDIC)* (2021).
[2] Ibid
[3] Ibid

☐ System Decomposition – Identify system processes by mapping operational tasks to system functions (Data Flow Diagram)

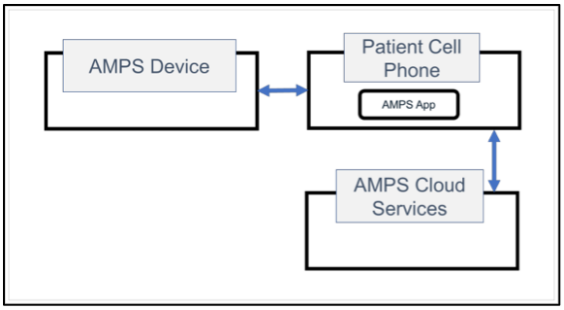
Goal: Map your routes through the system

☐ *Step 7: Develop a Data Flow Diagram (DFD) of your system*

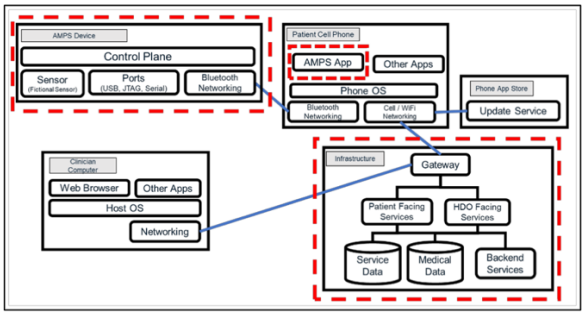
There are multiple ways to design a DFD, but we recommend the DFD3 standard. Begin by answering the following questions:

- What are the known components of the system?
- What components within your system connect to each other?
- What known third-party connections exist outside of your system's control?

From these questions, start to draw your diagram and gradually add additional components and sub-systems to the DFD depending on scope and time. Start at a high level and work your way down as seen in the below AMPS examples. Ultimately, these datapoints should come together to form a comprehensive map of your system.



Image[2]: High-Level DFD



Image[3]: Mid-Level DFD with Trust Boundaries

☐ *Step 8: Determine which system functions are associated with distinct operational tasks*

With the DFD of your system in hand, you can then link the system's operational tasks to specific system functions. When executing a specific task, what parts of the system are utilized? These include both assets and data flows between systems.

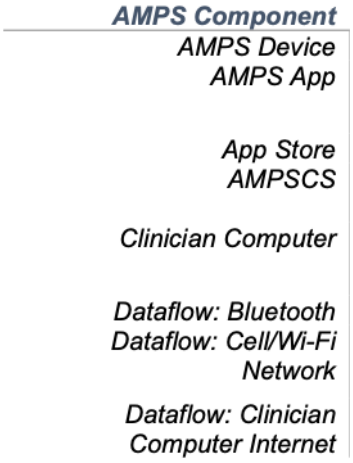
Mission Objective	Operational Task	System Function
Track patient's stroke risk	Collect sensor data	AMPS embedded sensors
Track patient's stroke risk	Store data in the cloud	AMPS cloud services
Securely share patient data	Store data in the cloud	AMPS cloud services

☐ Identifying Critical Assets

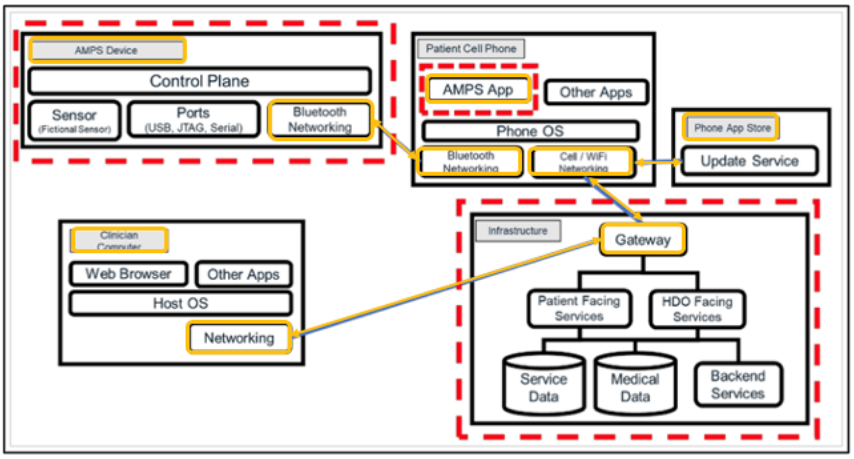
Now that you've done mission and system decomposition, you should have a much better idea of which system functions facilitate operational tasks that enable your mission. Using your DFD and the matrix from Part 7, you can now identify critical assets. Ask yourself the following questions:

- Which system assets and data flows are shared by multiple processes?
- What assets and data flows enable different system functions?
    - Establish mission dependencies.
- How does the failure of each operational task impact the system's mission objectives?
- What are downstream effects of taking each cyber asset offline?

In the example below, we've identified critical assets/components of the AMPS using our DFD, highlighting them in gold.



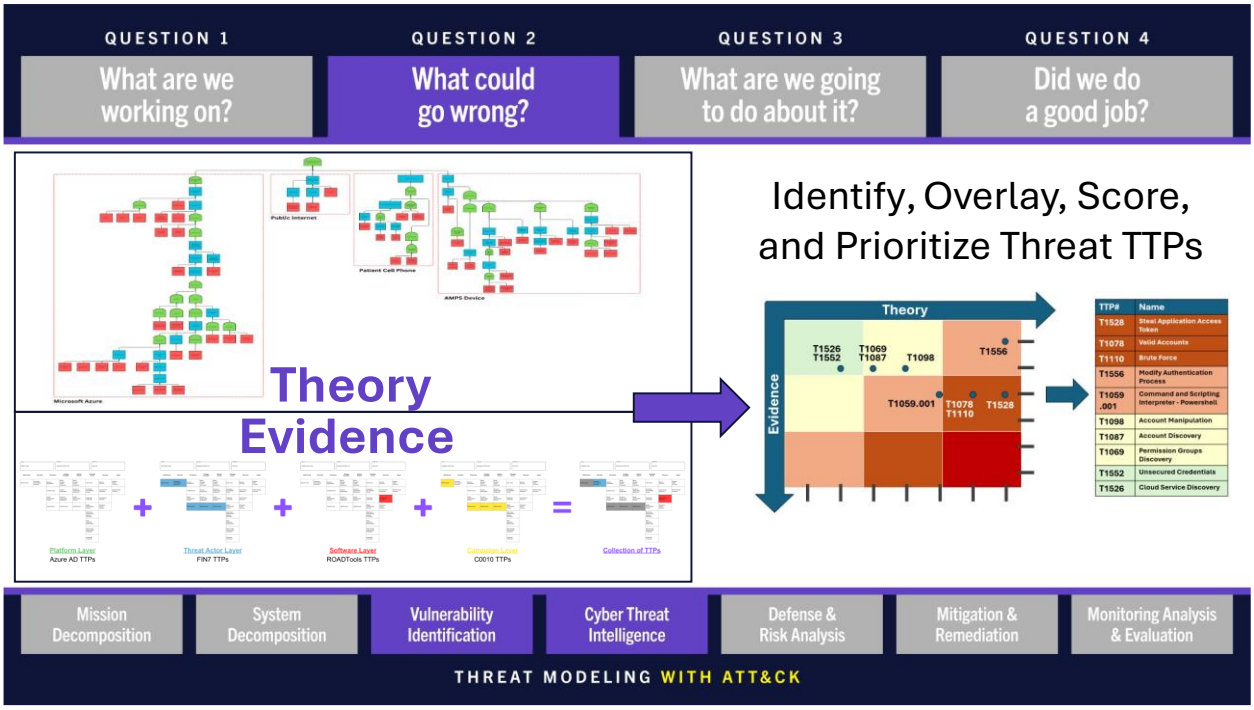
Image[4]: Critical AMPS System Components



Image[5]*: Mid-Level DFD with Trust Boundaries & ID-ed Critical Assets*

[4] Bochniewicz, Elaine, et al. "Playbook for Threat Modeling Medical Devices." *MITRE and the Medical Device Innovation Consortium (MDIC)* (2021).
[5] Ibid

Question 2: What could go wrong?



The process outlined in Question 1 derives critical assets within a particular system. In Question 2, we identify and prioritize threats to those assets. ATT&CK will serve as the framework through which we map and discuss threats to our system. While our analysis will go beyond the tactics, techniques, and procedures (TTPs) found within ATT&CK, its value as a language for detailing adversary behaviors makes it a central part of our approach. ATT&CK's widespread use within the CTI community and its comprehensive classification system allow us to draw upon existing threat data while still integrating additional threats not yet captured in public reporting.

For more information on MITRE ATT&CK, see these resources.

Theory vs. Evidence

Generally, there are two complementary approaches that can be utilized to perform threat analysis: **Theoretical Modeling**, and **Evidence-based Analysis**. Regardless of which threat modeling methodology you use to answer Question 2, you will need to strike a balance between these two approaches for your model to be effective. Both approaches handle different aspects of the threat landscape, directly addressing the potential ("*Could*" and "*Could not*") and the possible ("*Has*" and "*Has not*") threats that concern your system.

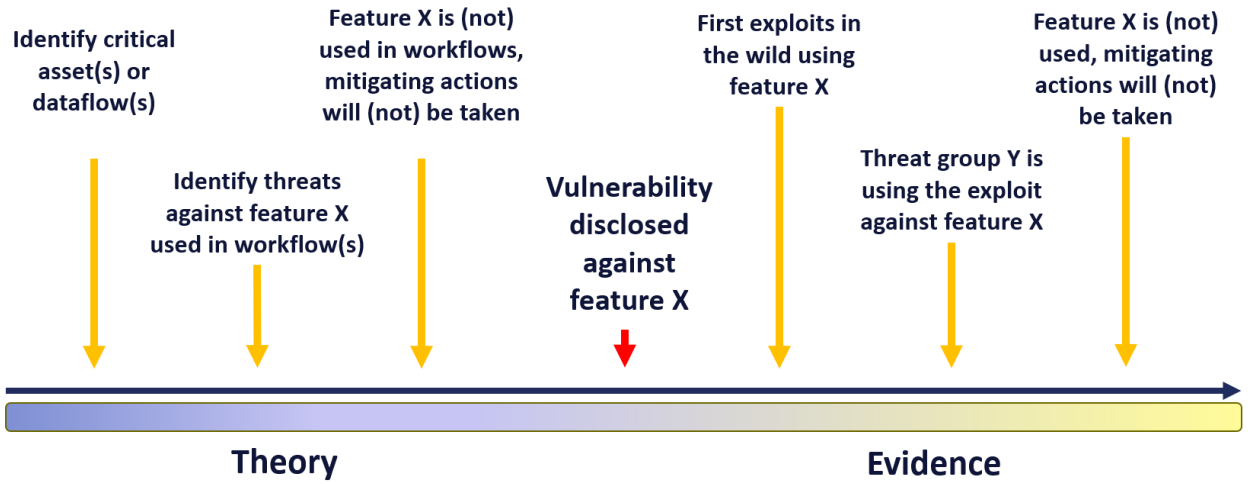
Evidence	Theory: Could not happen	Theory: **Could happen**
Has not happened	**Low Priority** *Lowest chance of active exploitation*	**Medium Priority** *Impact of threat exploitation should be carefully considered*
Has happened	**Medium Priority** *Review system mitigations related to threat for accuracy*	**High Priority** *System can be actively exploited if targeted*

The two axes on the above table represent the theoretical and evidence-based outcomes of a manifested threat.

- Theory describes threats that have potential to impact your system.
    - Theory-based threats are hypothetical threats. These include brainstorming conducted by your team, known exploits performed in a controlled environment, and hypothetical attacks that have not been leveraged by threat actors.

- Evidence describes documented threats that have been leveraged against other systems.
    - Evidence-based threats are observed threats. These include TTPs used to exploit technology platforms leveraged by your system, known exploits used by adversaries that target your industry, and malicious actions you've recorded within your system.

When considered together, these two approaches give a well-rounded view of a system's security posture, for both known and unknown threats.



Another way to consider how theory and evidence operate is in the context of vulnerability disclosure. The above timeline illustrates where and how theory and evidence support each other during the lifecycle of a zero-day. Inherently, theory-based modeling approaches tend to be more preparatory in anticipation of an unknown vulnerability, while evidence-based approaches tend to be more reactionary and respond to actualized threats in the wild.

In the following section, we will be modeling threats against the AMPS device to demonstrate a well-rounded theory and evidence approach. We will employ Attack Trees as our example threat modeling methodology that could be used to derive potential threats. Using our tree, we'll map theoretical and evidence-based threats an adversary might exploit to extract user information.

Theory

As we've discussed, a theory-based approach, regardless of which threat modeling methodology is used, identifies threats that have the potential to impact your system. We break this approach into three parts.

☐ Part 1: Structured Threat Brainstorming

Goal: Generate an unweighted list of threats against your system, based on the system analysis produced when answering Question 1

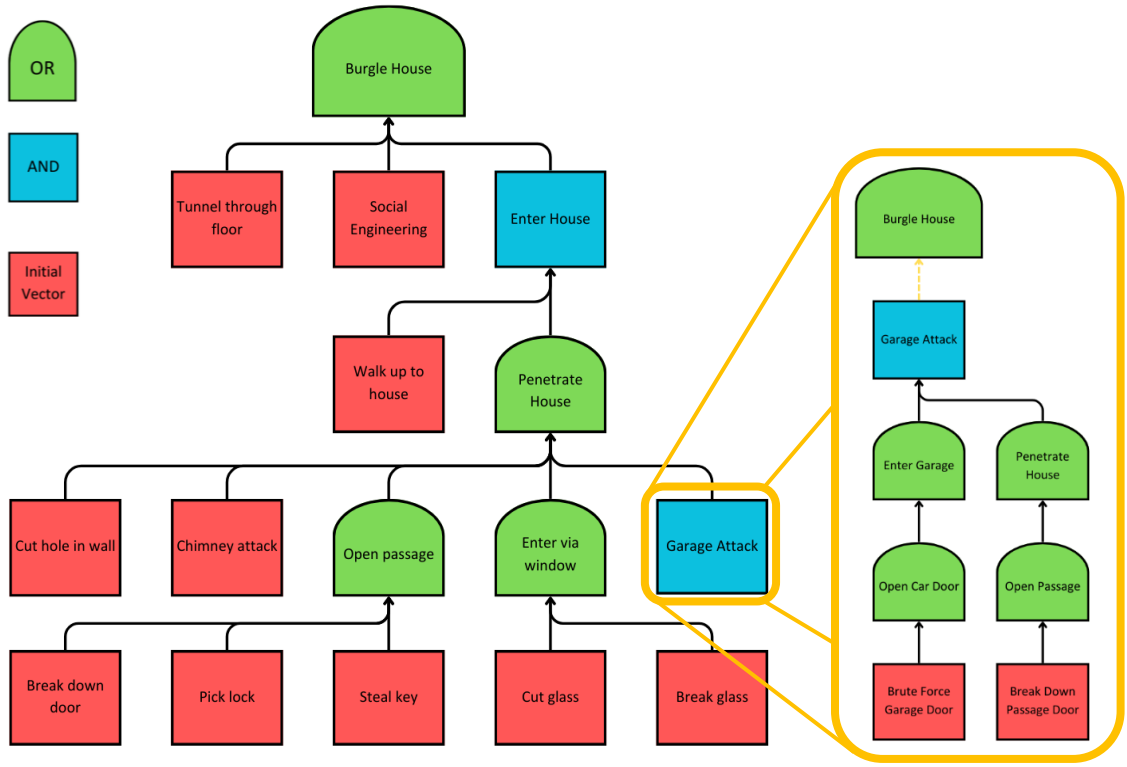
Mapping theoretical attacks on our system establishes the scope for our threat calculus. Much as we did in answering Question 1, we must think about the key assets we're trying to protect. We then start from a high level, identifying as many access vectors targeting our critical assets as we can, or at least a large representative sample. This wide aperture allows us to then hone our focus as we progress to cataloguing attacks based on real-world evidence in the next section. In this manner, we're able to capture attacks that are entirely possible but have yet to be observed in the wild, while also focusing much of our efforts on known vulnerabilities. While there are varying methods for building our catalogue of intrusions, we've chosen to leverage attack trees.

What is an attack tree?

An attack tree is a threat modeling technique that allows analysts to map the various ways in which an adversary could exploit a specific system to accomplish a specific set of goals. In the context of the AMPS, we've identified that a key mission objective is maintaining the confidentiality of the user's data. Our example attack tree will therefore aim at mapping the different pathways an adversary could take to access sensitive user information, namely their location.

Bottom-up attack trees

Attack trees typically represent the flow of attacker actions in two ways: top-down or bottom-up. The attack tree below relies on a bottom-up approach and will serve as our template moving forward. This tree captures the sequential pathways an attack could, and in some cases must, take to reach its objective. Regardless of the attacker's intention, any adversary seeking to exploit a given system must achieve these intermediate goals. In this manner, the tree is agnostic towards the attacker's subsequent goals.

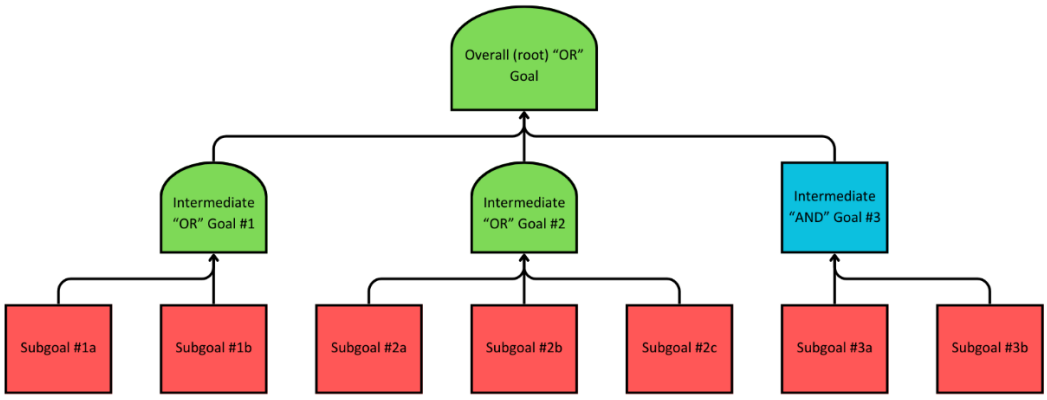


Image[7]: Example of Bottom-Up Attack Tree and One of its Isolated Sub-Trees

Here we see a theoretical attack tree for a thief attempting to burgle a house. The thief has several potential avenues for achieving their goal. Some are more complex than others, requiring multiple steps. Some constitute entire sub-trees of their own, such as the "Garage Attack." Each attack has its associated characteristics: the cost of the attack, the complexity, the likelihood of success, the time needed to execute it. Each of these will influence the attacker's actions and therefore influence where mitigation strategies should be deployed.

The origin point of the tree is the kernel, or root node, the ultimate objective of the attacker that sits at the top of the tree (in the example above, the root node of the tree is "Burgle House"). The attacker works their way towards that objective by satisfying the intermediate goals that branch out from the root node. Each branch represents a different exploitation strategy that can or must be employed to achieve the ultimate objective. In some cases, a particular strategy (branch) must be executed to allow another strategy to move forward.

[7] Ingoldsby, Terrance R. "Attack tree-based threat risk analysis." *Amenaza Technologies Limited* (2010).



Image[8]: Attack Tree design language

The arrow-shaped OR nodes within the tree represent goals that can be achieved by any of the goals below them (here, Intermediate Goal 1 OR 2 OR 3). The flat bottom AND nodes, similarly, are fulfilled by the goals listed beneath them. All these goals (here, Subgoal 3a AND Subgoal 3b) must be fulfilled to progress. The square subgoals represent the actions that must be taken to achieve their final goal.

Using our knowledge of the system we codified responding to Question 1, we now need to brainstorm potential attacks that could be launched against the critical assets we identified. We will do this using an attack tree. Initially, the nodes within the tree can be conceptual in nature. In the later steps, these will become more granular.

Visualizing attack trees

To visualize these attack trees, we used (and recommend using) MITRE Engenuity's Attack Flow Builder (see below), but there are several other simple and complex tools you can use to build your attack trees. The easiest approach is to use a common tool like Microsoft Word or PowerPoint. The graphic design tool Canva is another great, easy-to-use option (any graphic design software can work as well). For more formal tools capable of complex analysis, there are a few options:

- SecurITree, developed by Amenaza Technologies, is purpose-built for attack tree analyses and allows for the addition of detailed attributes to different attack paths, risk metrics, and adversary personas.
- The AT-AT (Attack Tree Analysis Tool) allows users to develop and analyze attack scenarios in much the same way.
- AttackTree by Isograph similarly allows for attack tree modeling and additional threat analyses beyond the capabilities of a basic visualization tool.

All of these are viable options for crafting attack trees of your own.

[8] Ingoldsby, Terrance R. "Attack tree-based threat risk analysis." *Amenaza Technologies Limited* (2010).

Part 2: Critical Path Analysis

Goal: Find commonalities between threats produced during brainstorming and identify critical paths or components in your system.

In this step, just as we mapped system processes to critical assets in Question 1, we're taking the theoretical attacks we've brainstormed and associating them with critical paths and components.

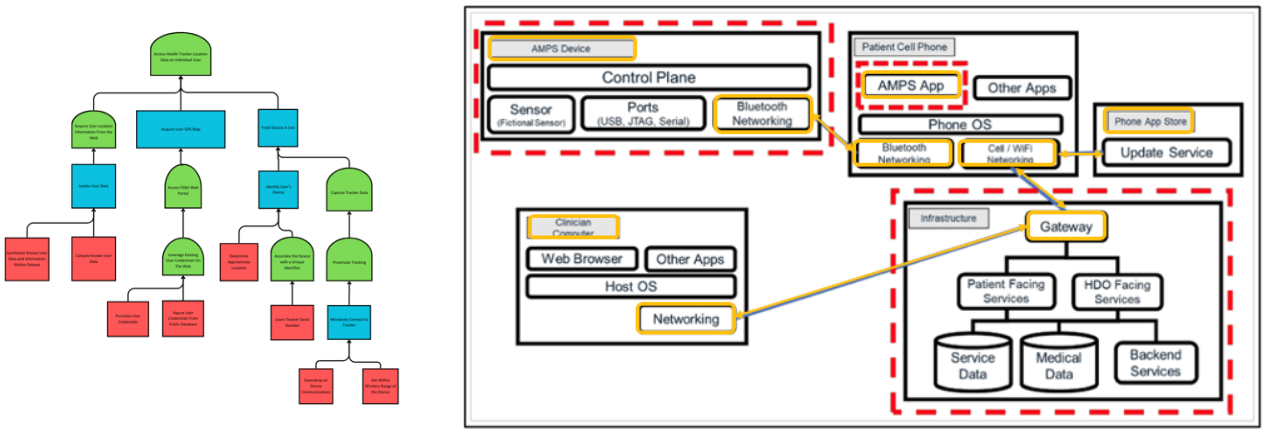


Image: Bottom-up Simple Attack Tree against AMPS location information alongside Mid-Level DFD of relevant critical assets

As we establish these associations between threats and assets, we'll begin distilling our theoretical threats. This exercise will clarify how steps in an attack are associated with one another, determining which attacks must be executed and in what order. It will also verify whether certain steps in an attack are still possible once mapped onto specific assets within the system.

In the following example, we've created an attack tree and populated it with theoretical threats against our AMPS device. In Question 1, we said collecting and securely storing patient data was essential to our product. We've therefore made the goal of our attack tree stealing patient sensor data, specifically user location data. We've spoken with our team, trawled academic literature, reviewed blog posts by industry professionals, and watched presentations by security experts to create an initial set of theoretical threats to our device. Another resource we reviewed was MITRE's EMB3D threat knowledge base, which worked great to break down the AMPS device by its properties and the specific threats to each. For more help brainstorming insider threat behaviors, take a look at the Center's insider threat knowledge base. Taken together, all this research gives us an initial list of threats we can then associate with our critical assets. See the AMPS attack tree below for an example of the compiled theoretical threats against our critical assets.

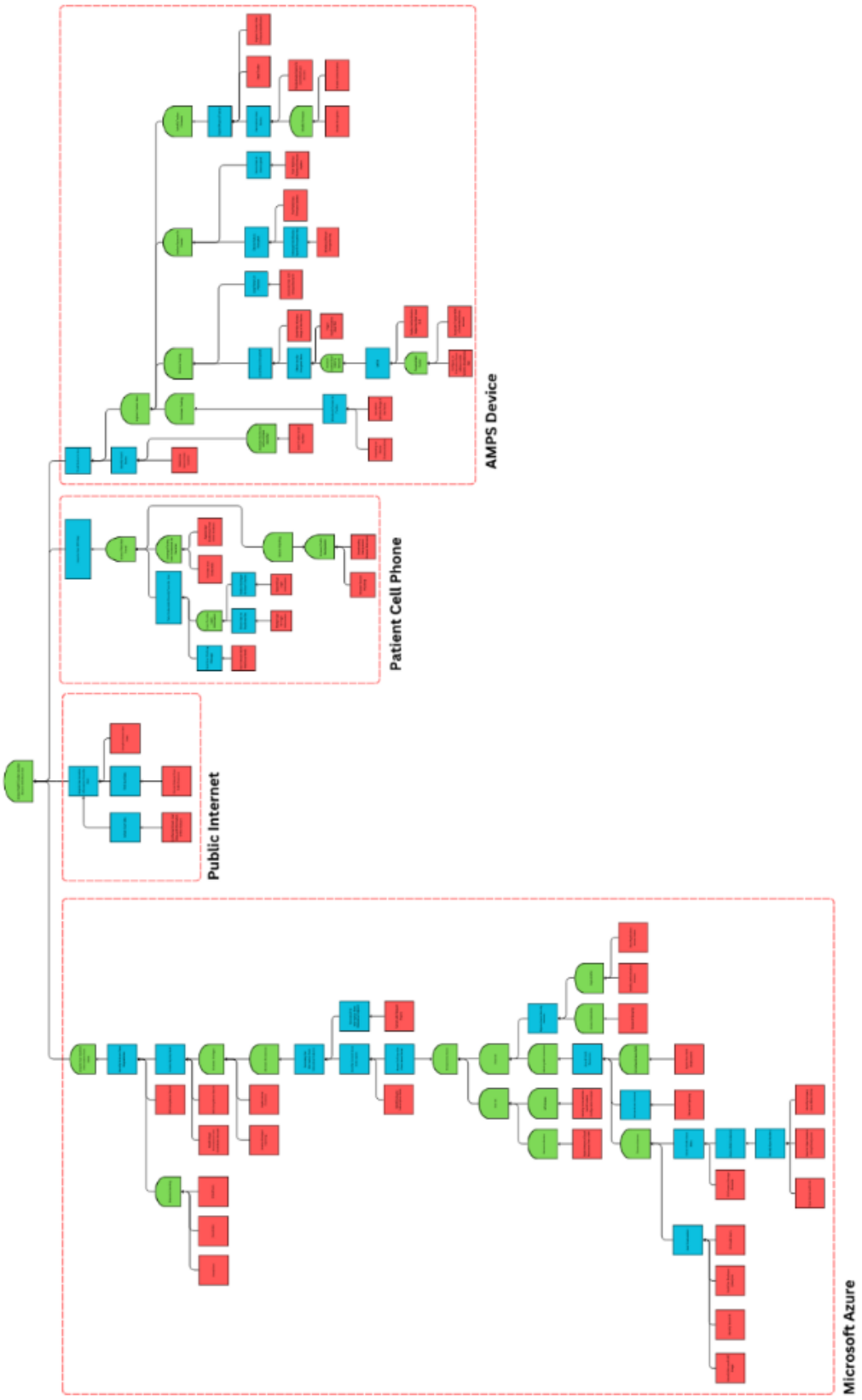
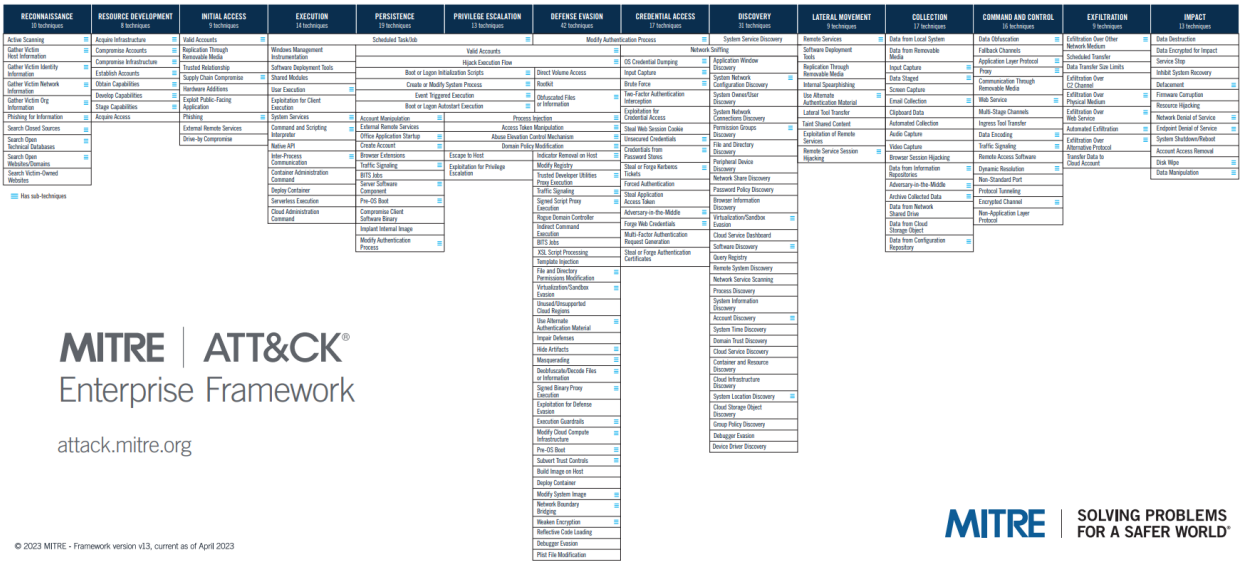


Image: Example AMPS attack tree mapped to our critical assets, rotated to fit this page

Part 3: Translating Attack Tree Concepts into ATT&CK TTPs

Goal: Use ATT&CK as a common language to describe adversarial behaviors against your system



Image[9]: Example of an ATT&CK Framework

Now that we've built out our attack tree, clarifying our language and invoking specific system data exchanges and assets, we can begin cataloguing the ATT&CK TTPs needed to facilitate those attacks on each critical path and component. These datapoints will constitute the core of our attack tree and link our results from this theoretical exercise to the results of our evidence-based analysis later.

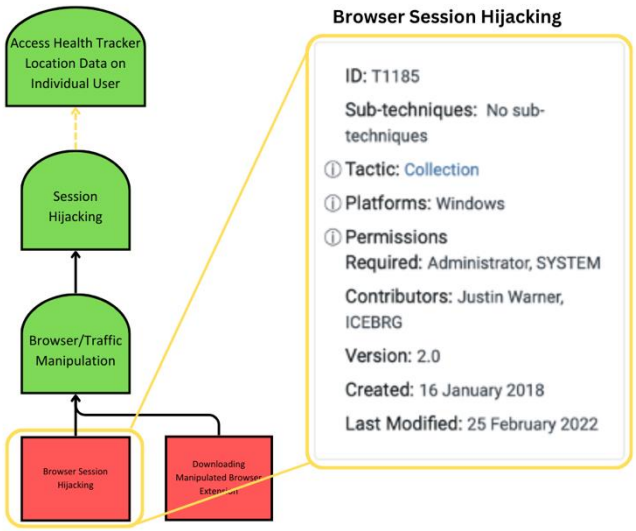
This step is essentially the manual translation of Part 2's conceptual attack steps into tangible ATT&CK TTPs. We recommend using Decider to assist in these translations. This tool allows you to either filter for specific tactics, platforms, and data sources that will direct you towards the appropriate TTP, or search key terms related to your attack concept in the search bar to derive the appropriate TTP. When comparing your Part 2 attack tree concepts to existing ATT&CK TTPs, consider adding nodes to your attack tree for any TTPs you may not have thought of.

Below is an example of how a theoretical attack can be aligned with a TTP (Browser Session Hijacking T1185).

During our search for threats relevant to the AMPS device, we determined that one of the vectors (branch of the tree) an attacker could use to access user location data was by accessing their web portal. We determined that one potential vector for gaining access to the user's portal was by stealing their log-in credentials. This can be done using an activity characterized as Session Hijacking in ATT&CK.

[9] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

Ultimately, we will be integrating these threats into a singular tree using the Center's Attack Flow tool and directly linking them to our critical assets. Attack Flow integrates seamlessly with ATT&CK. Threat actor actions represented as nodes on the tree can be linked to specific TTPs. Furthermore, additional contextual elements such as attack characteristics, assets, data types, conditions, and references can be added to each node of the tree. With Browser Session Hijacking (T1185) identified as one of our theoretical exploits, we can now associate that specific node on the tree with T1185, thereby pulling in all the data that's been associated with that exploit. Not all the threats you identify will be directly tied to TTPs, but these threats should still be included in your tree and will still inform the response you develop in Question 3.



An example of the AMPS attack tree and all associated TTPs can be found below.

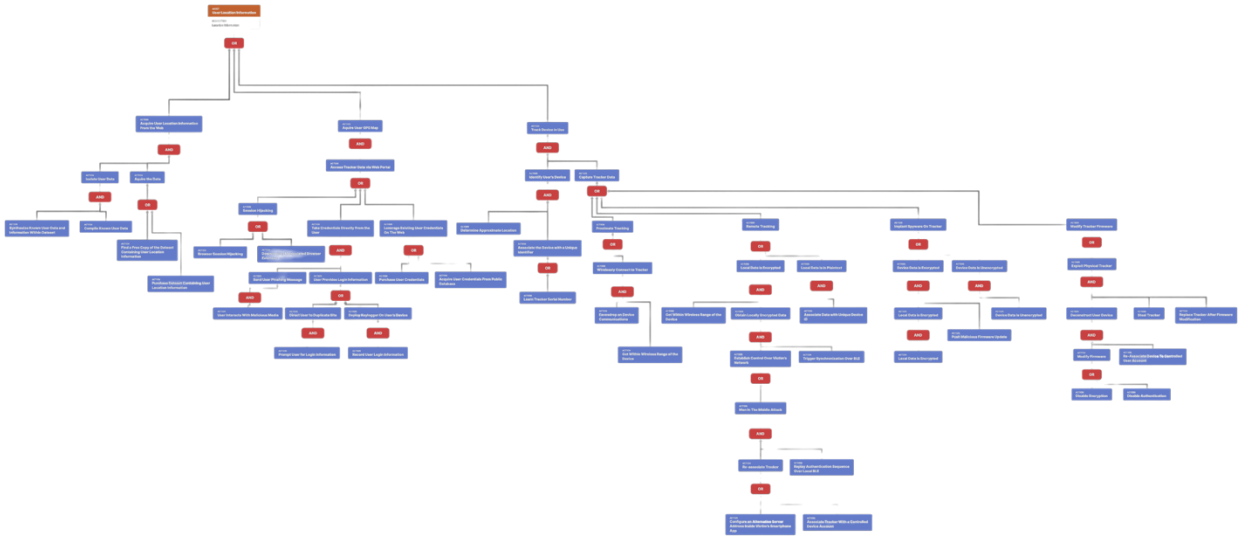


Image: Example AMPS Attack Tree Converted into Attack Flow

Evidence

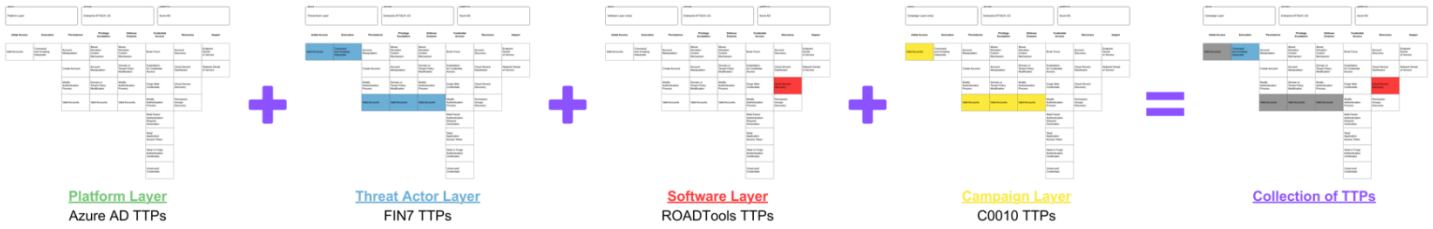
The previous section focused on a theory-based approach using attack trees. In this section, we will cover the evidence-based approach to complement our theoretical tree and aid in identifying additional TTPs for consideration in the tree. Evidence is derived by attacks observed in the wild and reported on by legitimate sources. The MITRE ATT&CK team reads open-source reports published by these sources and associates adversarial behavior with a TTP. Sources for these TTPs are different from those previously used to build the theory-based attack tree, which is why the complementary approach of theory and evidence is crucial. We will use the TTPs derived in this section to add to the attack tree in the previous section. We recommend considering TTPs derived by four types of observed behavior.

1. TTPs used against your Technology Platform(s)
2. TTPs used by Threat Actor(s) targeting your Industry
3. TTPs used by Software used maliciously against your Industry
4. TTPs used by Campaign(s) targeting your Industry

Throughout this section, we break down each type of observed behavior and demonstrate how to use the TTPs describing this behavior in your attack tree. We will continue to use AMPS in all examples.

Multiple technology platforms were identified in our attack tree. For the purposes of this paper, however, we will only be using observed TTPs related to the cloud platform (Azure) branch of the attack tree.

As we walk through this section and explain how to generate TTPs from each of the four types of observed behaviors above, we will start to compile a consolidated list of TTPs pertinent to branches of our tree (in this case the Azure branch). These TTPs will be compiled in the form of ATT&CK Navigator Layers. The figure below shows the process of stacking the multiple ATT&CK Navigator Layers derived from each category of data. The information gathered in this section will also support scoring in the following section.



Image[10]: Layered Steps to Form Collection of TTPs

The observed TTPs in these layers may not have been previously used to achieve the goal we are analyzing in our attack tree (user location data). This is expected. Often, intrusions go through your company to access your business partners or customers. Although your company, or others in your industry, may not have been the desired end target in these reported incidents, you were an intermediate target and the TTPs used in these "leap frog" intrusions against your industry or tech platform can be used to target you in the future. Thus, we include them in our observed TTP layers.

[10] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

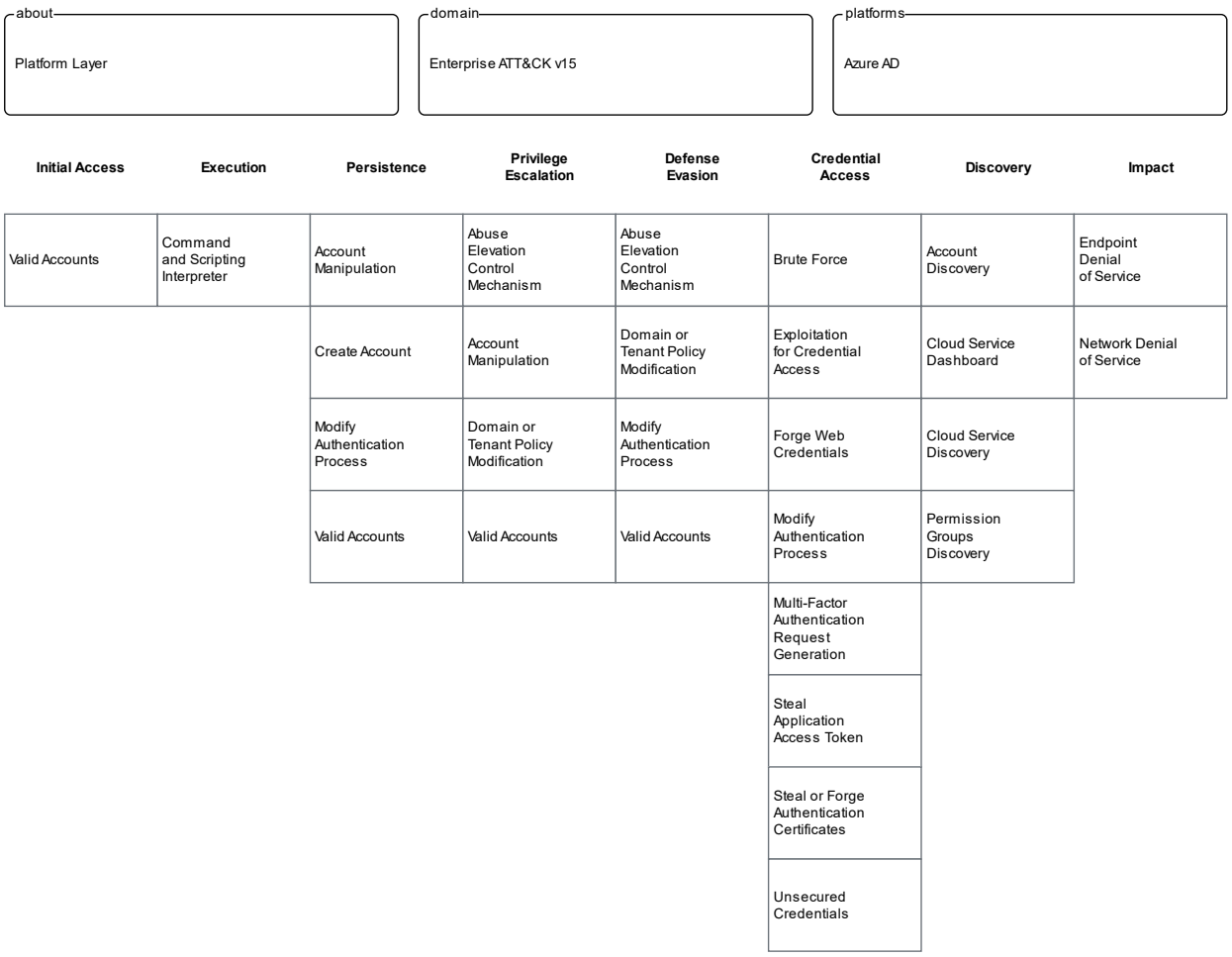
☐ Layer 1: Technology Platform TTPs

Goal: Compile a list of TTPs that have been used to target your tech platform

To characterize the observed threats targeting your system, we recommend starting with techniques targeting your specific technology platform. This information will be used to prioritize threats in your attack tree later.

Types of observed CTI data vary by company, depending on which commercial data you subscribe to or which public datasets you leverage. As a best practice, if the data is available, internally generated observed threat data targeting your network and technology platforms should be incorporated. For the purposes of our example, the fictitious team evaluating AMPS doesn't pay for any CTI data and only has publicly available data at its disposal. A good starting place for any team, regardless of budget, is ATT&CK Navigator. This tool allows you to filter mobile, enterprise, or industrial control system matrices by technology platform. Our theory-based attack tree is already broken down into technology platform branches, and the focus is on generating observed TTPs one branch at a time. Navigator will generate an ATT&CK matrix with TTPs targeting your technology platform that have been observed in the wild. ATT&CK version 14.1 has the following platform filters: macOS, Windows, Linux, Azure AD, PRE, Containers, Office365, SaaS, Google Workspace, and IaaS. We recommend adding TTPs (or Navigator Layers) derived from your commercial data or data generated internally to this technology platform Navigator layer. This additional data will help capture more observed TTPs used against your technology platform.

Below is an ATT&CK Navigator view showing the TTPs linked to Azure AD. Throughout this evidence section, we will down-select from these base-layer TTPs.



Image[11]: Example ATT&CK Navigator Layer for Azure AD

☐ Layer 2: Threat Actor (TA) TTPs

Goal: Compile a list of TTPs that have been used by a threat group/s targeting your industry

If time permits, we also recommend generating threat profiles to characterize the adversaries, or groups, that are likely to target your industry and therefore your system. This information will also help in prioritizing threats in your attack tree later.

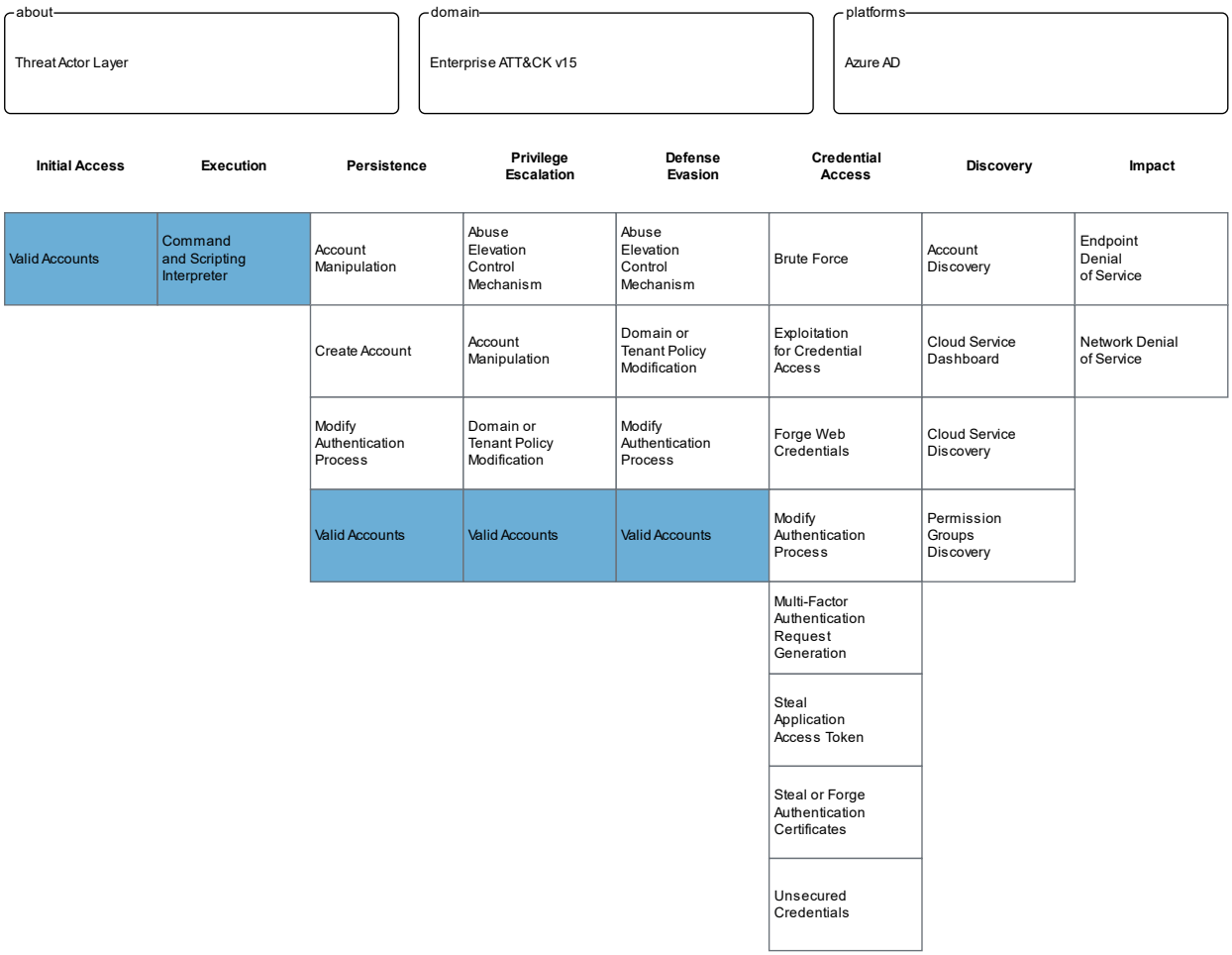
To get started with threat actors that are relevant to your organization, consider any threat actors that have been known to be a concern in the past, or have been mentioned recently as a concern to your organization. It is always a good idea to consider threat actors that have previously been a threat to your organization since they are known to you. Ask your stakeholders if there are any TAs they are concerned with too.

The ATT&CK Groups knowledge base can be a good starting point for any team. The Groups page gives an overview of all the TAs reported publicly. Although many CTI vendors have their own naming structure, MITRE Groups is an attempt at combining these TAs under a single naming convention. On this page, you can "CTL + F" to look for groups relevant to you. Some focus areas to search for might

[11] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

be location (i.e., United States, Iran, China) or industry (i.e., financial, government, retail); both searches help to narrow down threat actors important to your organization. Also keep an eye out for when these groups were active. Groups that have not been active recently might not be useful to your organization, but this is an internal decision that needs to be made based on your organization's needs. Be sure to keep these dates in mind as they will affect the scoring in the next section.

A Navigator layer exists on each Group's page. Use this layer to generate a list of TTPs for each TA you identified. Below is an ATT&CK Navigator example for FIN7 that highlights the TA's TTPs in blue. This threat actor was chosen by searching "medical" on the ATT&CK Groups page, which identified this group as previously targeting our industry's "medical equipment."



Image[12]: Example ATT&CK Navigator Layer for FIN7

This is our first down-select from the technology platform layer. Additional TAs and the following layers will provide more. If you have more time to spend on this layer, once you've finished using the ATT&CK Groups page, you should look at threat actors in the news that are potentially relevant to your industry. If your organization subscribes to commercial data, search those databases or use Threat Intelligence Platforms (TIPs) available to you. An example of this can be found in Appendix A. Another good starting point for teams on a budget is the APT Groups and Operations Google Sheet. This

[12] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

spreadsheet consists of a list of threat actors by country as well as their name and aliases, operations associated, origin, toolset/malware utilized, a description of their motives/goals, and targeted industries.

This spreadsheet contains community-derived information. Because it is a living spreadsheet with various people making edits, it allows for a more real-time approach in terms of updates that can be helpful to organizations focusing on a specific threat actor. Ultimately this resource is another opportunity to find more evidence-based TTPs associated with the actor.

One final open-source resource is the Thai CERT database. This database allows you to search for threat actors by country, sector targeted, motivation, or key word. Once you've identified TAs of concern, compare these to the aliases on the ATT&CK Groups page ("CTL + F" search for name) and consider using any resulting group's Navigator Layer.

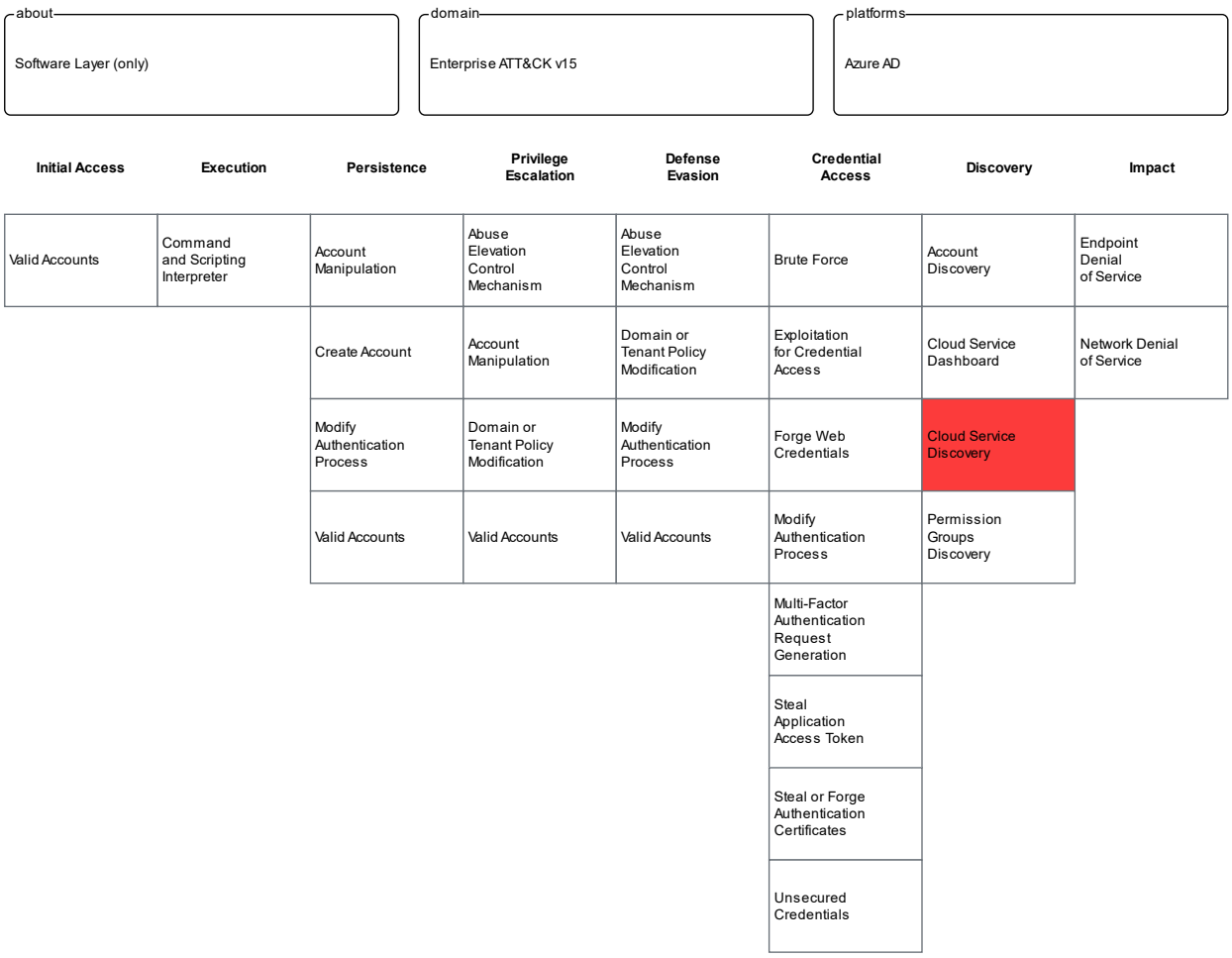
☐ Layer 3: Malicious Software TTPs

Goal: Compile a list of TTPs that have been used for the execution of publicly available (malicious) tools

The next step will follow a similar process to the steps above. To start, an organization should always compile internal data first. This can be done by utilizing datasets within any TIPs you use as well as any previous threats your company has seen. Starting with the known and building on the new data allows for a more exhaustive list of TTPs while ensuring company-specific data is considered.

After reviewing internal and commercial data, use the ATT&CK Software page similarly to how we used it for the TA layer. In this scenario you will use it to build a list of TTPs used by malicious software targeting your specific technology platform. This will be done by accessing the ATT&CK Software page and using "CTL + F" to search for your technology platform.

In our case, we search "Azure," which results in two findings of software: AADInternals and ROADTools. For the sake of this example, the team will focus on ROADTools. We recommend including all software pertaining to your platform, or just specific software you find most applicable; you will have to make this decision based on your needs and time. During this step, remember that ATT&CK software is not just compromised or malicious software, but also commercial, open-source, built-in, or publicly available software that could be used by a defender, pen tester, red teamer, or adversary conducting "living off the land" techniques. Each Software page comes with a Navigator Layer. The ROADTools ATT&CK Navigator layer can be seen below in red.



Image[13]: Example ATT&CK Navigator Layer for ROADTools

☐ Layer 4: Campaign TTPs

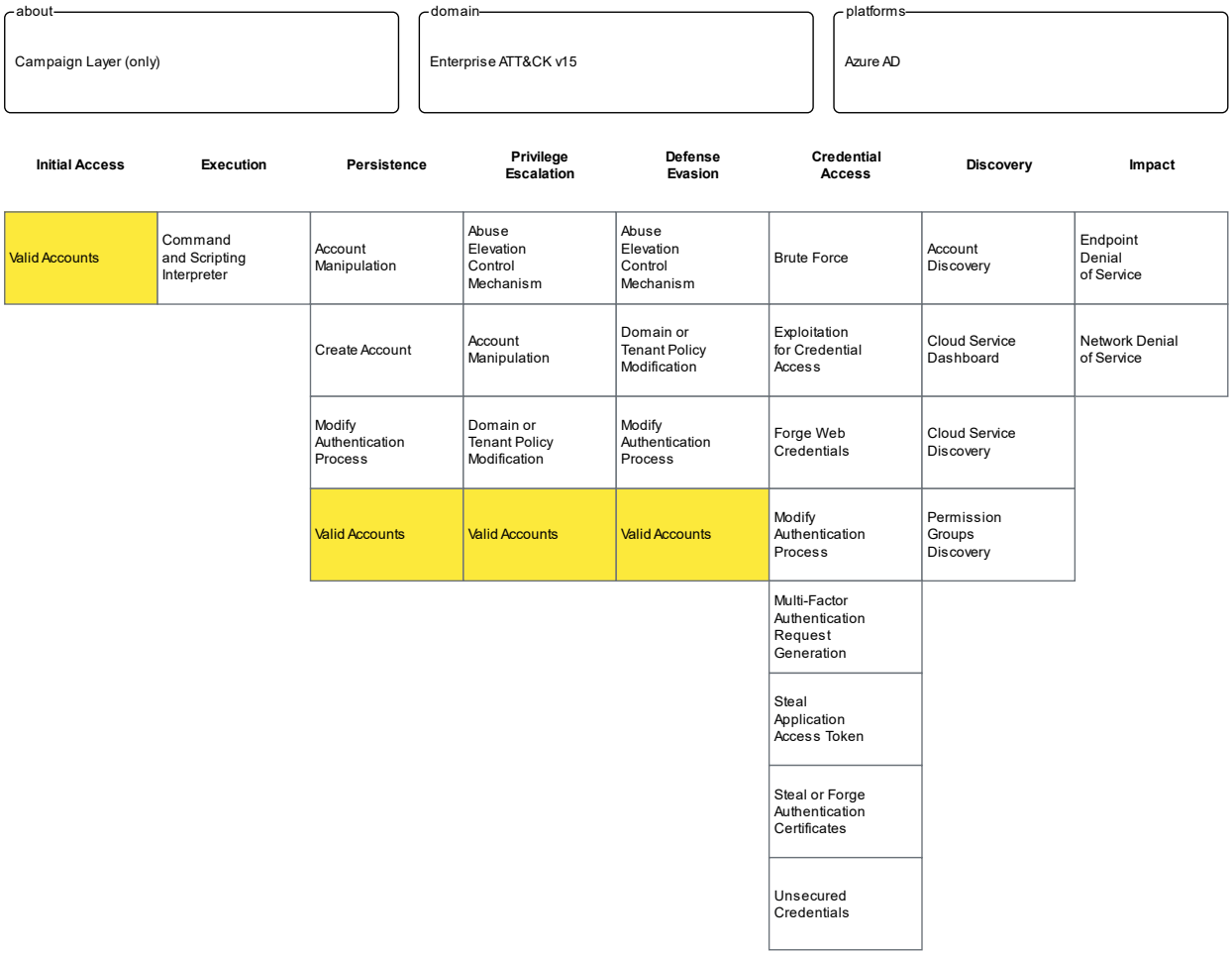
Goal: Compile a list of TTPs that have been used in a campaign targeting your industry

To provide a more detailed picture, if your organization has the time, it is recommended you research campaigns that might be applicable to you. This can be done in various ways similar to the previous layers. First, any campaigns recently reported on that are of concern to your organization should be included. It might also make sense to include any data from previous campaigns that targeted your organization as well as data from tools used internally. Once this data has been considered and added, the team should use the ATT&CK Campaigns page for further research. Focus on campaigns targeting your specific industry. These can be searched by using "CTL + F" on the ATT&CK campaign page. During this step, be cognizant of the timing of these campaigns, since some may be too old to be useful. Only your organization can know which campaigns they find useful, but keep these dates in mind as they will affect the scoring in the next section.

For the AMPS device, we focused on one of the campaigns related to healthcare, specifically C0014. In many cases, this campaign might be considered not recent enough to be relevant, but for the sake

[13] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

of this example we will use it, despite the reported date being in 2022. The ATT&CK Navigator Layer below highlights the TTPs relevant to this campaign in yellow.



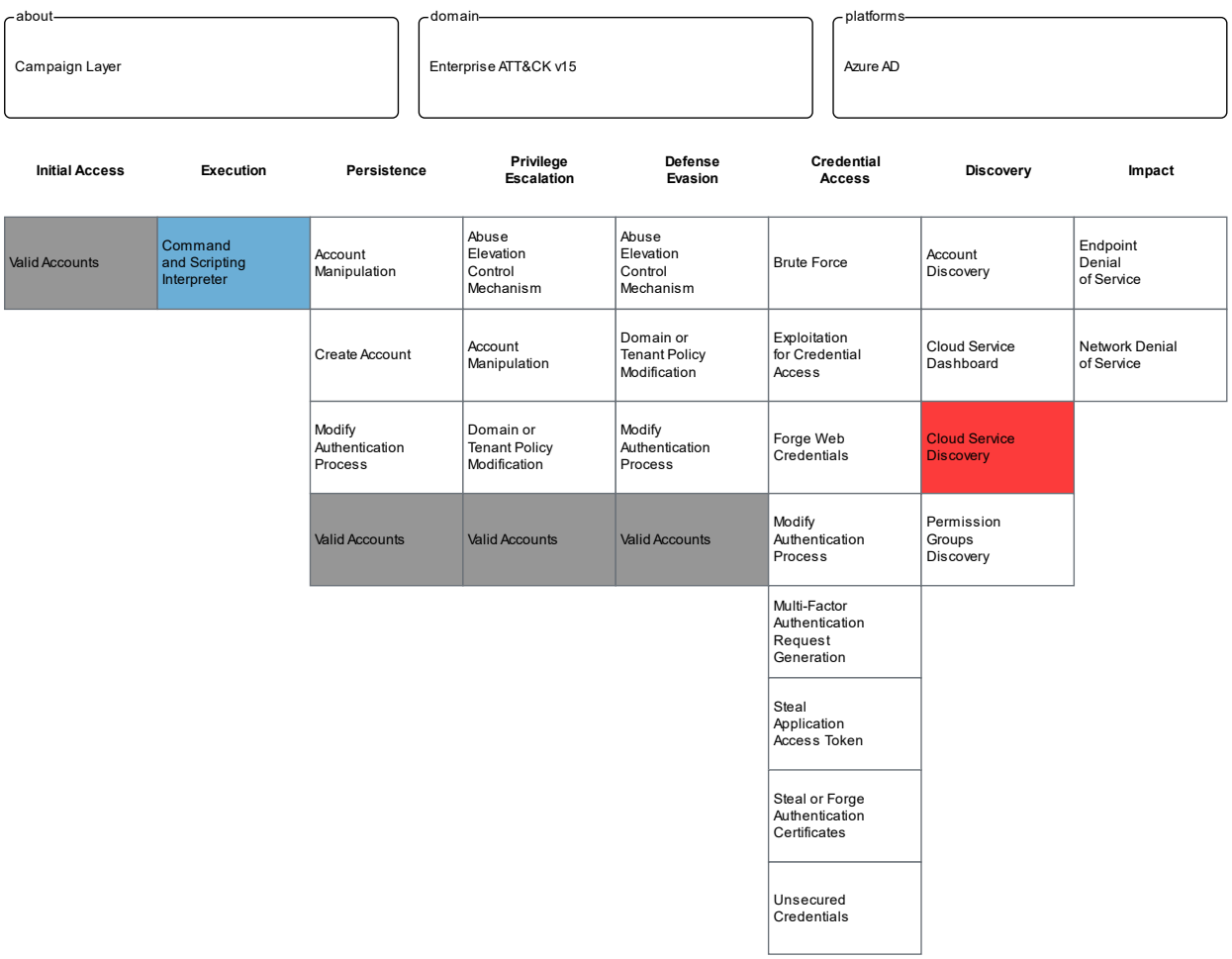
Image[14]: Example ATT&CK Navigator Layer for C0014

☐ Compile All CTI Layers and Compare to Theory-Base Attack Tree

Goal: Compile a list of TTPs that your system will most likely face

Right now you have a list of TTPs, in the form of ATT&CK Navigator Layers, that have been observed against technology platforms in your tree. Take those lists and overlap them all using Navigator. The overlap between layers can provide some insight for prioritization. The example below shows a combination of all layers used as examples above. The blue TTPs show those used by threat actors targeting your industry, the red TTPs signify the TTPs used by malicious software targeting your industry, the yellow highlights the TTPs used by campaigns targeting your industry, and grey shows any overlap between multiple layers.

[14] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

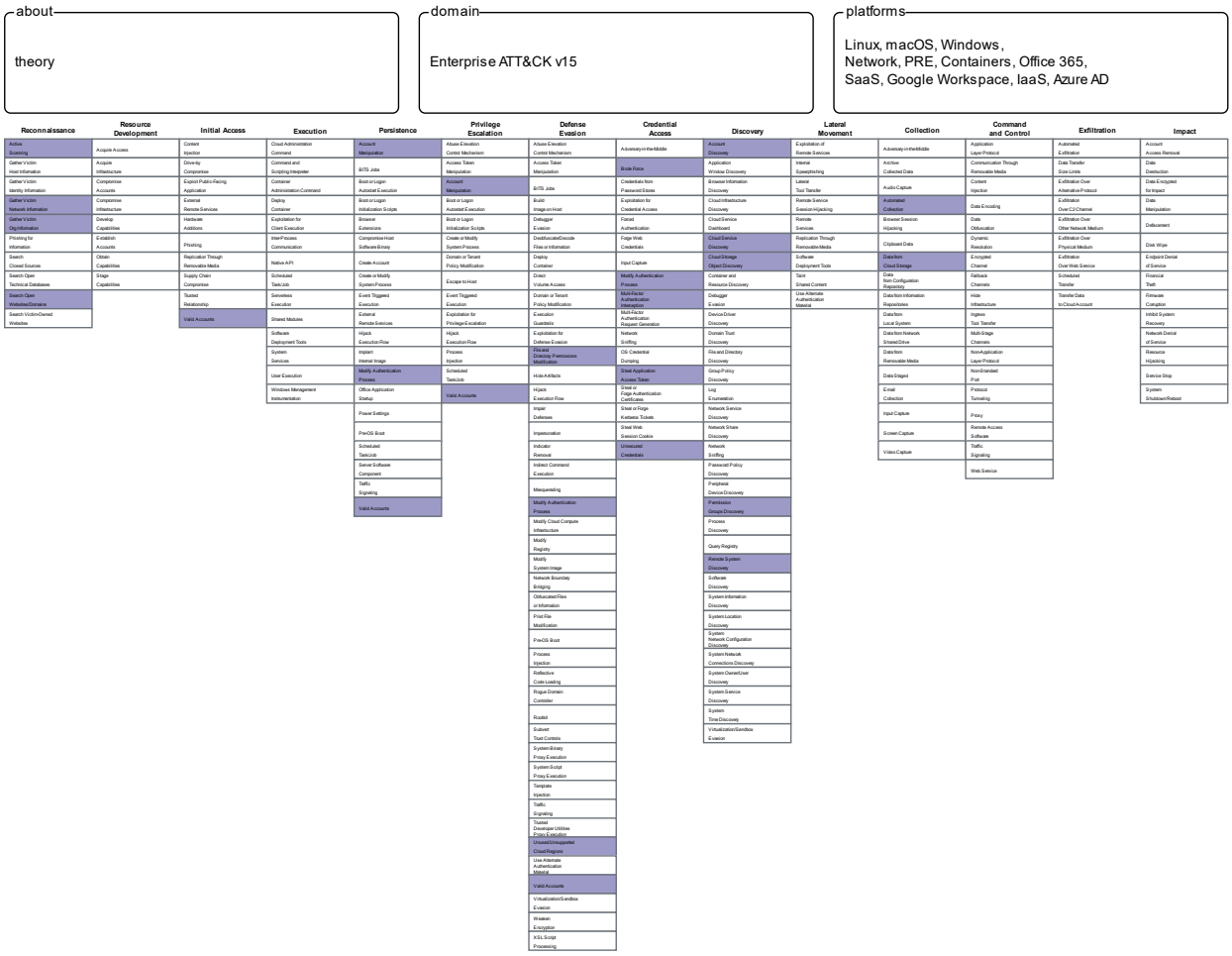


Image[15]: Example ATT&CK Navigator Layer for Combined Layers

Compare these TTPs to those in your theory-based attack tree. Since these TTPs are all related to the Azure branch of the attack tree, we will focus there. In practice, you will make one combined overlay for each technology platform branch of your tree.

To apply this to our current example, we will take our attack tree branch centered around Azure and map the steps back to ATT&CK techniques, as seen in the Navigator Layer below

[15] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

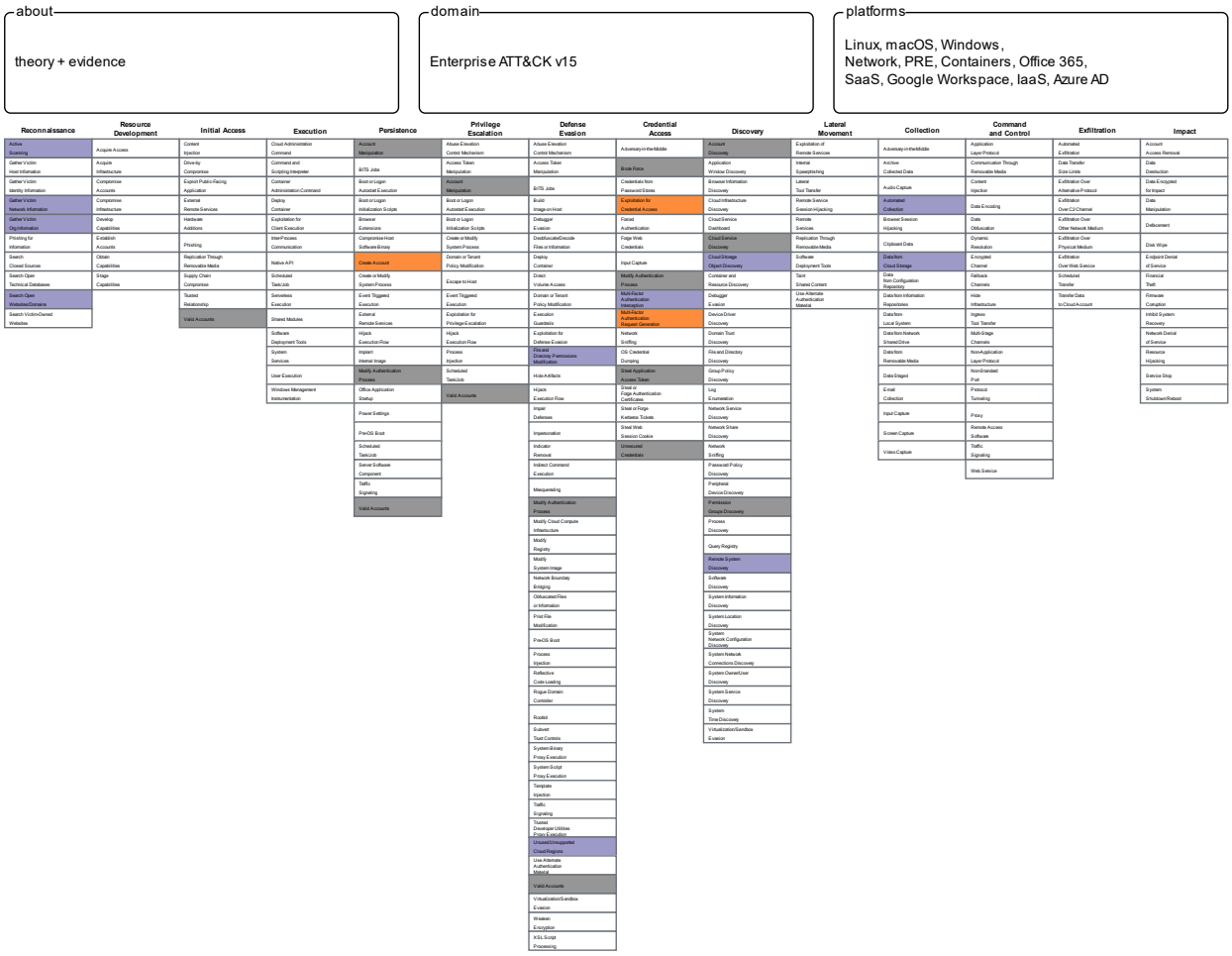


.

Image[16]: Example ATT&CK Navigator Layer for Azure Theory Branch

This Navigator Layer is now placed on top of our overall evidence layer (above); the TTPs that are supported by theory and evidence are highlighted in orange.

[16] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023



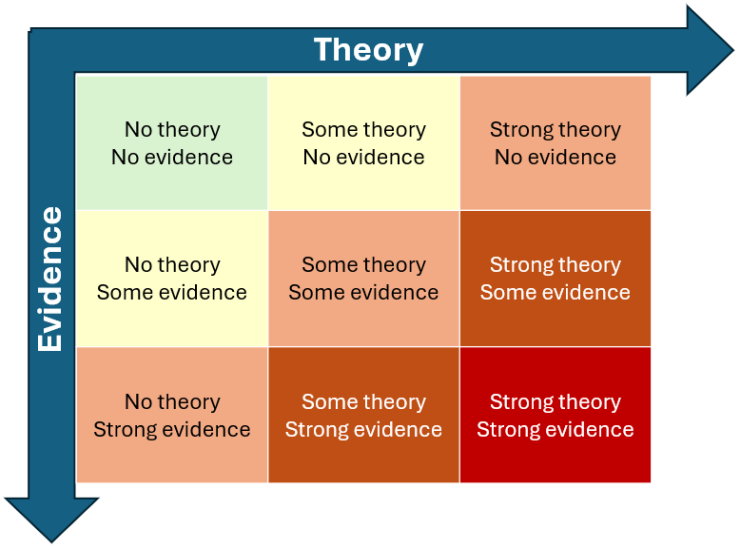
Image[17]: Example ATT&CK Navigator Layer for Azure Theory Branch Overlayed with Evidence Layers

Your next step is to evaluate the techniques that are not overlapping to see if they have a place in the Azure branch of the attack tree. Once you've added any new and relevant evidence-based TTPs to your branch, the resulting list of evidence and theory attack tree TTPs will be used in the next section.

[17] MITRE. "MITRE ATT&CK Enterprise Framework." *MITRE Corporation*, 2023

Scoring the Catalogue of Threats to Your System

This step lets us calculate the level of threat associated with specific attack vectors and TTPs. The end goal of this step is to prioritize which threats to mitigate in Question 3. Note, if you are limited on time you can skip this step and proceed directly to Question 3 with your long list of TTPs. However, conducting this scoring step might save you more time in Question 3 by enabling you to focus only on high-threat TTPs.

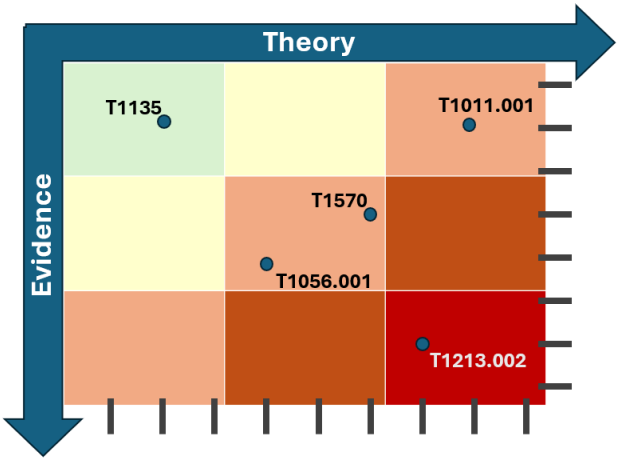


Revisiting the ideas presented in the introduction to Question 2, we can organize identified TTPs into different priority categories depending on the strength of their individual theory and evidence factors. These categories are not meant to be a strict numerical ranking – rather, they should be used as an aid to help prioritize your time and effort while evaluating mitigations and countermeasures.

Given a particular TTP identified by your overlay of theory and evidence, consider some of the following factors to help guide your prioritization of TTP data. Note that **this list is non-exhaustive**, and you may wish to incorporate other factors specific to your use case.

Factors indicating stronger Theory	Factors indicating stronger Evidence
TTP has been hypothesized in a research paper	TTP has been used by a threat group targeting your industry
TTP has been demonstrated in a technical lab	TTP has public reports of execution using publicly available (malicious) tools
TTP has known, publicly available tools for execution	TTP has been used in a campaign targeting your industry within the last 90 days
TTP has associated vulnerabilities (CVEs) applicable to your tech platform(s)	TTP has been used in a campaign targeting a tech platform you use within the last 90 days
TTP is associated with accessing a critical cyber asset	TTP is associated with a vulnerability/CVE disclosed within the past 30 days
TTP is associated with a critical system choke point identified in system diagrams	TTP has been used against your tech platform in the past
TTP is associated with a critical system choke point identified in threat analysis	

The more factors that apply for either theory or evidence, the further you move in the table right or down, respectively. The simplest form of this analysis assigns an equal value to all factors (i.e., a weight of 1). However, you may find that some factors should be treated with more importance to suit your prioritization needs. For example, you may give TTPs associated with external system boundaries (i.e., external network connections) extra weight to prioritize developing mitigations for system entry points.



The result will manifest like the diagram shown. TTPs are assigned a theory-evidence score, which places them at a point in the table. Thresholds can be individually adjusted for both theory and evidence to determine how large or small to make the sectors in the table. For example, in industries that utilize newer or more specialized technology, there may be less available evidence to consider in your threat overlay. Consequently, you may choose to weigh individual pieces of evidence higher for other industries.

Example scoring

Consider **TTP: T1011.001** – Exfiltration Over Other Network Medium: Exfiltration Over Bluetooth

Assume the adversarial goal in this case is to steal sensitive patient data. One avenue would be to go directly to the source – the AMPS device itself.

T1011.001 describes activity where "*Adversaries may attempt to exfiltrate data over Bluetooth rather than the command-and-control channel. If the command-and-control network is a wired Internet connection, an adversary may opt to exfiltrate data using a Bluetooth communication channel.*" The AMPS device has been designed with Bluetooth in mind, as it needs to pair with a phone.

Several Bluetooth vulnerabilities have been documented in the literature, but we will choose to focus on one named SweynTooth.[18] SweynTooth is a collection of vulnerabilities in certain Bluetooth Low Energy (BLE) chipsets, with a range of impacts ranging from crashes to security bypass. Perusing the website dedicated to this vulnerability, we can come to the following conclusions on the strength of **theory factors**:

- The TTP has been **hypothesized** in the writeup (beyond hypothesized, in fact)
- The TTP has been **demonstrated** (there is proof of concept code against multiple devices)
- The TTP has **known tools** for execution (there is proof of concept code)
- SweynTooth is a Bluetooth vulnerability and therefore applies to this TTP
- Patient data is a **critical cyber asset** for this device (which the TTP directly affects)
- The Bluetooth connection between the AMPS device and the patient phone is a link that crosses a trust boundary on the DFD (and is therefore a **critical link**)

[18] https://asset-group.github.io/disclosures/sweyntooth/

- This TTP is present in attack tree branches that directly access the device, but there are other ways to get patient data (e.g., compromising their online account). Ergo, this **may or may not** be considered a choke point from a threat analysis standpoint.

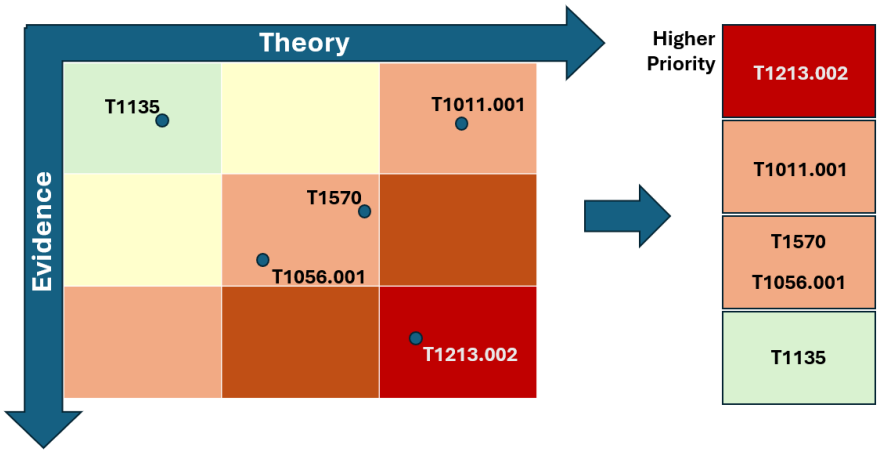
On the theory side, the above culminates in **6/7 factors** applying here, indicating **strong supporting theory** for this TTP.

With respect to evidence, we see a much different story manifesting:

- Threat groups operating against the healthcare industry have generally **not** been targeting Bluetooth (caveat – at the time of writing)
- There **are** several reports of Bluetooth exploits being leveraged in the wild
- Similar to the first point, there are very few **campaigns** leveraging Bluetooth in the wild, and by extension, very few campaigns targeting this industry and tech platform
- While Bluetooth is generally regarded as insecure, there have not been any major vulnerability disclosures over the past 30 days (at the time of this writing)

On the evidence side, the above culminates in **1/5 factors** applying here, indicating **little or weak supporting evidence**. Together, the theory and evidence place this TTP toward the upper right on the figure, which gives this TTP a medium priority under normal weighting.

To reiterate, this step is not meant to produce a definitive first-to-last ranking of TTPs – rather, it serves to quickly prioritize where to focus your efforts when considering countermeasures and mitigations in Question 3. Therefore, once you are done sorting TTPs, sort the boxes, rather than the individual TTPs themselves, for priority. Returning to the example figure, this would result in the following prioritization scheme.



Depending on your priorities, you may choose to sort the categories of TTPs differently if your concerns align more with theory or with evidence; i.e., you may choose to prioritize the center box higher than the top right box if you are more worried about strength of evidence than strength of theory.

Example Scoring TTPs within AMPS's Azure Attack Tree Branch

The following table summarizes the TTPs identified during the Theory and Evidence activities presented earlier in this section. We've sorted the table into three columns – Theory, Evidence, and both, to track which activity each TTP was derived from.

Theory	Evidence	Theory & Evidence
(T1595.002) Active Scanning – Vulnerability Scanning	(T1136) Create Account	(T1526) Cloud Service Discovery
(T1590.001) Gather Victim Network Information – Domain Properties	(T1212) Exploitation for Credential Access	(T1078) Valid Accounts
(T1591.002) Gather Victim Org Information – Business Relationships	(T1621) Multi-Factor Authentication Request Generation	(T1098) Account Manipulation
(T1591.004) Gather Victim Org Information – Identify Roles		(T1110) Brute Force
(T1593) Search Open Websites/Domains		(T1528) Steal Application Access Token
(T1134) Access Token Manipulation		(T1552) Unsecured Credentials
(T1098.001) Account Manipulation Additional Cloud Credentials		(T1087) Account Discovery
(T1222) File and Directory Permissions Modification		(T1069) Permission Groups Discovery
(T1535) Unused/Unsupported Cloud Regions		(T1556) Modify Authentication Process
(T1110.003) Brute Force – Password Spraying		(T1556) Modify Authentication Process
(T1111) Multi-Factor Authentication Interception		
(T1552.005) Unsecured Credentials – Cloud Instance Metadata API		
(T1619) Cloud Storage Object Discovery		
(T1530) Data from Cloud Storage		
(T1119) Automated Collection		
(T1018) Remote System Discovery		
(T1580) Cloud Infrastructure Discovery		

To keep the rest of this example concise, we have elected to only score the TTPs listed under the "Theory and Evidence" column. However, scoring can (and should) be applied to all identified TTPs.

Theory factor scoring

1. *TTP has been hypothesized in research paper(s)*
2. *TTP has been technically demonstrated in a published setting (lab, presentation, etc.)*
3. *TTP has known, publicly available tools for execution*
4. *TTP has associated vulnerabilities (CVEs) applicable to your tech platform(s)*
5. *TTP is associated with accessing a critical cyber asset in your system*
6. *TTP is associated with a critical system choke point identified in system diagrams*
7. *TTP is associated with a critical system choke point identified in threat analysis*

TTP #	*Tactic*	*TTP Name*	*#1*	*#2*	*#3*	*#4*	*#5*	*#6*	*#7*	*Total*
T1078	**Initial Access**	Valid Accounts	X	X	X	X	X	X		6
T1098	**Privilege Escalation**	Account Manipulation	X	X			X		X	4
T1110	**Credential Access**	Brute Force	X	X	X		X	X	X	6
T1528	**Credential Access**	Steal Application Access Token	X	X	X	X	X	X	X	7
T1552	**Credential Access**	Unsecured Credentials		X	X					2
T1556	**Credential Access**	Modify Authentication Process	X	X	X	X	X	X	X	7
T1087	**Discovery**	Account Discovery		X	X				X	3
T1526	**Discovery**	Cloud Service Discovery		X	X					2
T1069	**Discovery**	Permission Groups Discovery		X	X				X	3

Some notes on the above:

- Datapoints for Factor 1 encompass TTPs that are theoretically possible but have yet to be demonstrated. Threats were primarily identified from academic publications and industry publications.
- Sources for Factor 2 often pull from academic and industry publications, but these exploits have been corroborated by testing. Presentations by security professionals at conferences and online are another valid source for this information.
- Satisfying Factor 3 entails tracking down sources that link the identified TTP with existing tools. For this example, Azure red teaming reports were a key source in identifying known tools associated with specific TTPs.
- Entries for Factor 4 were determined by searching through existing CVE repositories for CVEs specifically tied to Azure and Microsoft products.
- Entries for Factor 5 were identified by reviewing our attack tree and determining whether a TTP directly targeted critical assets.
- Entries for Factor 6 were identified by examining our original DFD. Chokepoints or interests that represent key information bottlenecks within the system were identified.
- Entries for Factor 7 were identified in much the same way as Factor 6, but in this case choke points were identified within the system attack tree as lynchpins within a larger adversary campaign.

Evidence factor scoring

1. *TTP has been used by a threat group targeting your industry*
2. *TTP has public reports of execution using publicly available (malicious) tools*
3. *TTP has been used in a campaign targeting your industry within the last 90 days*
4. *TTP has been used in a campaign targeting a tech platform you use within the last 90 days*
5. *TTP is associated with a vulnerability/CVE disclosed within the past 30 days*
6. *TTP has documentation of previous use against your tech platform*.

TTP #	*Tactic*	*TTP Name*	*#1*	*#2*	*#3*	*#4*	*#5*	*#6*	*Total*
T1078	**Initial Access**	Valid Accounts	X	X				X	3
T1098	**Privilege Escalation**	Account Manipulation	X					X	2
T1110	**Credential Access**	Brute Force	X	X				X	3
T1528	**Credential Access**	Steal Application Access Token	X	X			X	X	3
T1552	**Credential Access**	Unsecured Credentials	X					X	2
T1556	**Credential Access**	Modify Authentication Process					X	X	2
T1087	**Discovery**	Account Discovery	X					X	2
T1526	**Discovery**	Cloud Service Discovery		X				X	2
T1069	**Discovery**	Permission Groups Discovery	X					X	2

Some notes on the above:

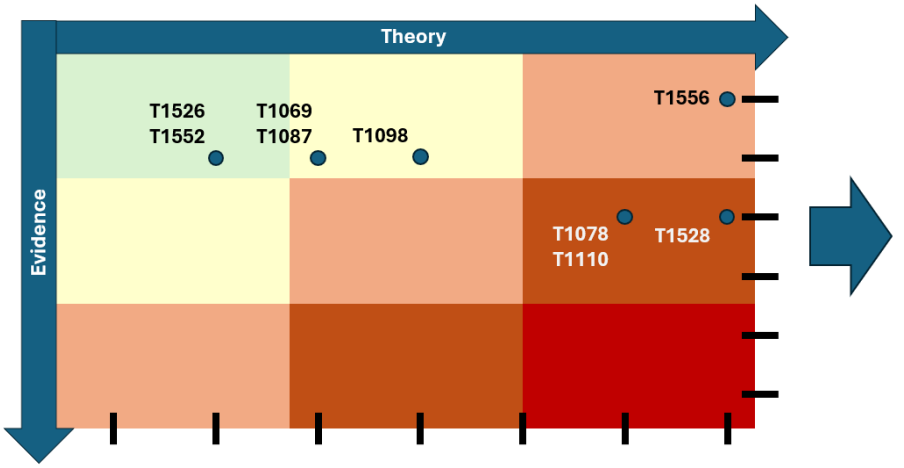
- Entries for Factor 1 were determined by searching the Groups page on the ATT&CK website. Relevant groups were identified by searching for the keyword “healthcare,” where their TTP lists were cross-referenced with entries in the table.
- Entries for Factor 2 were determined by searching the relevant TTP entries in ATT&CK for related software artifacts applicable to Azure.
- Entries for Factors 3 and 4 were determined by searching campaigns on the ATT&CK website targeting Azure. At the time of writing, there are no known campaigns occurring within the last 90 days against Azure. While there have been campaigns targeting healthcare in the past, they have largely focused on denial of service and ransomware outcomes,[19] which fall outside of the scope of the TTPs we are evaluating.
- Entries for Factor 5 were determined by a keyword search for “Azure” on the CVE website. While there are multiple Azure CVEs at the time of writing, none are related to the TTPs.

[19] https://www.microsoft.com/en-us/security/blog/2023/03/17/killnet-and-affiliate-hacktivist-groups-targeting-healthcare-with-ddos-attacks/

- Entries for Factor 6 were taken directly from the ATT&CK Navigator Overlay presented in Evidence Layer 1 detailing TTPs relevant to the Azure platform.

It is important to note that Factors 3, 4, and 5 are all considered with restricted time windows, as allowing all instances of a TTP may lead to over-scoring based on "stale" information; i.e., a campaign that occurred two years prior, while informational, does not carry the same urgency as a campaign actively happening within the last month.

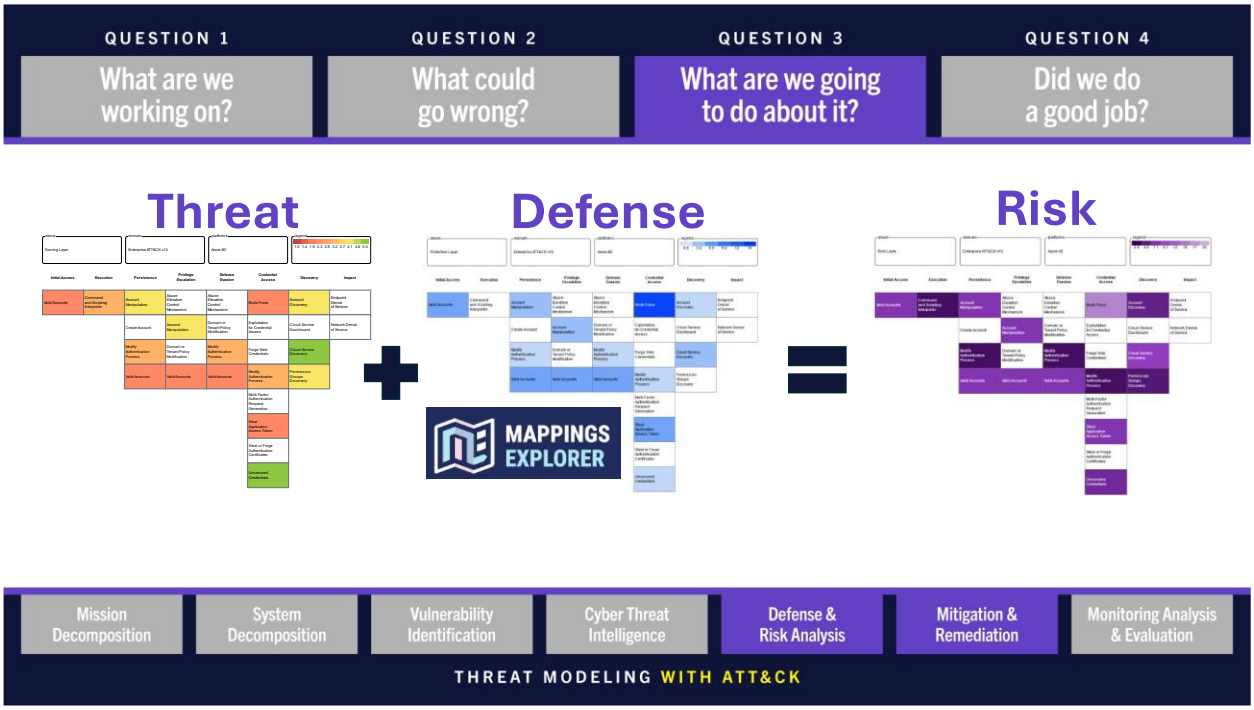
After scoring, the TTPs can be placed on a heatmap overlay, then sorted by grouping from highest to lowest priority. The following figure illustrates the outcome of this process. Points on the heatmap with multiple listings represent TTPs that achieved the same score. Note that in this example, T1556 could have their positions exchanged, depending on whether your priorities align closer to Theory or Evidence factors.



TTP#	Name
T1528	Steal Application Access Token
T1078	Valid Accounts
T1110	Brute Force
T1556	Modify Authentication Process
T1098	Account Manipulation
T1087	Account Discovery
T1069	Permission Groups Discovery
T1552	Unsecured Credentials
T1526	Cloud Service Discovery

As a reminder, this example only scored TTPs that appeared during both Theory and Evidence investigations. When creating a full threat model, it is important to consider all TTPs for completeness.

Question 3: What are we going to do about it?



Now that we have a prioritized list of TTPs our adversaries will likely use against our specific tech platform(s), we need to identify how our tech platform(s)' existing security measures mitigate them. This section will provide a guide for using the Center's Mappings Explorer[20] website to identify which existing security capabilities within your environment are mapped to the threats you're concerned about. If the Explorer's existing mappings don't fit your needs, this section will also introduce a process for mapping security controls and capabilities, native to a technology platform or mapping framework, to ATT&CK TTPs. These resources can be used to understand, assess, and record the real-world threats that security controls within your technology platform are able to mitigate. Using these Mappings, we can prioritize defensive investments against high-priority TTPs targeting our technology platforms. Continuing with the AMPS example in Question 2, we will see which of the TTPs identified within our Azure attack tree branch are mitigated by leveraging the Azure mapping within Mappings Explorer.

Mappings Explorer Overview

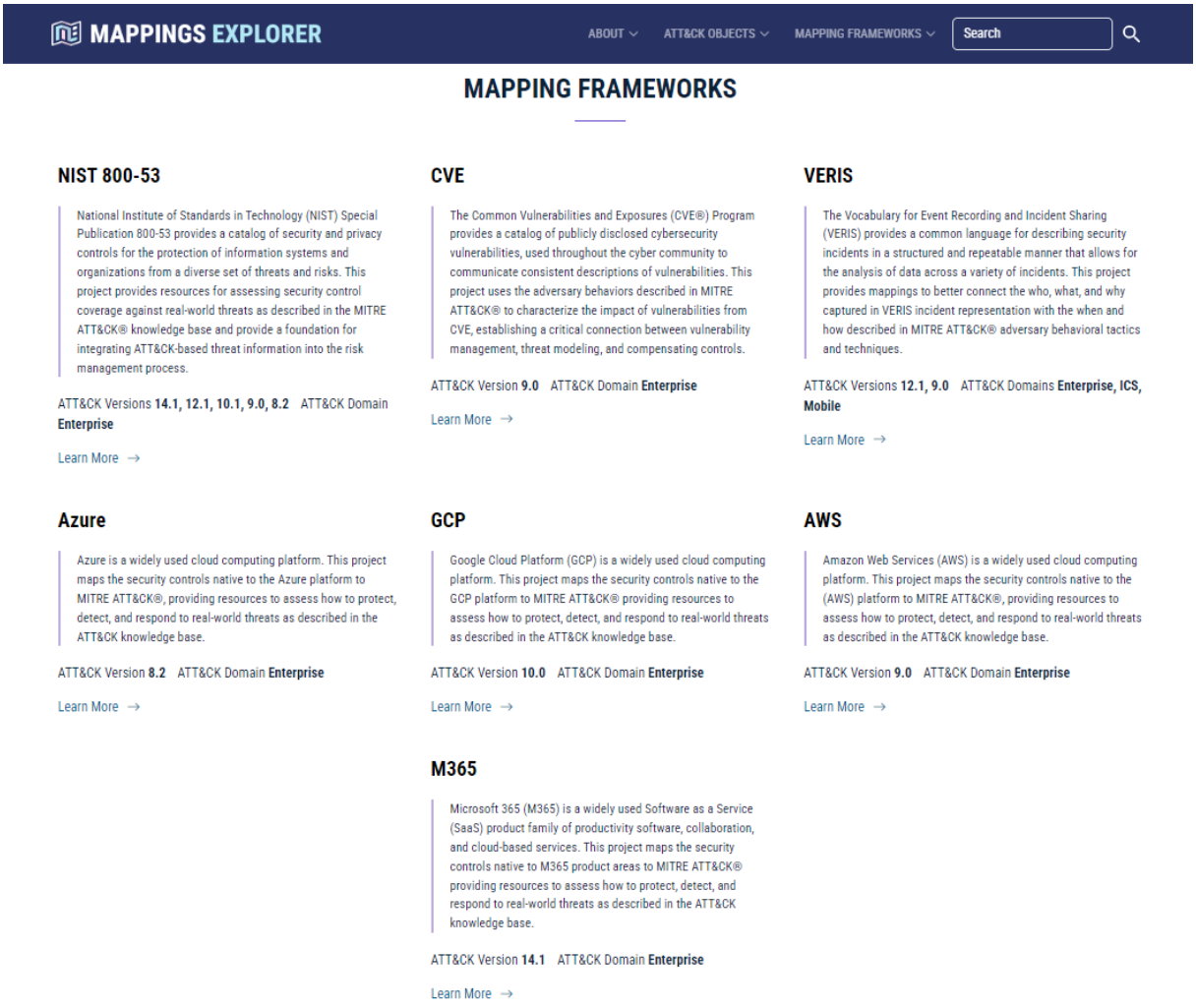
The Center provides a collection of mappings connecting security capabilities to the ATT&CK framework through Mappings Explorer. This website hosts a collection of open, independently developed mapping products, tools, and resources. These mappings form a bridge between the threat-informed approach to cybersecurity (Question 2) and the traditional security controls perspective.

[20] The Center for Threat-Informed Defense (2024), *Mappings Explorer*: https://center-for-threat-informed-defense.github.io/mappings-explorer/

Mappings Explorer enables cyber defenders to understand how security controls and capabilities map onto adversary behaviors catalogued in the ATT&CK knowledge base. The website presents security control mappings and threat and mitigation data in user-friendly ways. This enables the exploration of adversary techniques and the corresponding mapped capabilities across platforms and frameworks.

The mappings provided in Mappings Explorer are designed to provide independent data on which native security capabilities are most useful in defending against specific adversary TTPs. You will need to decide what types of capability functions are applicable for implementation in your environment and meet your threat mitigation needs.

The security capabilities of the following frameworks mapped to ATT&CK are freely and openly available:



Mapped Frameworks in Mappings Explorer

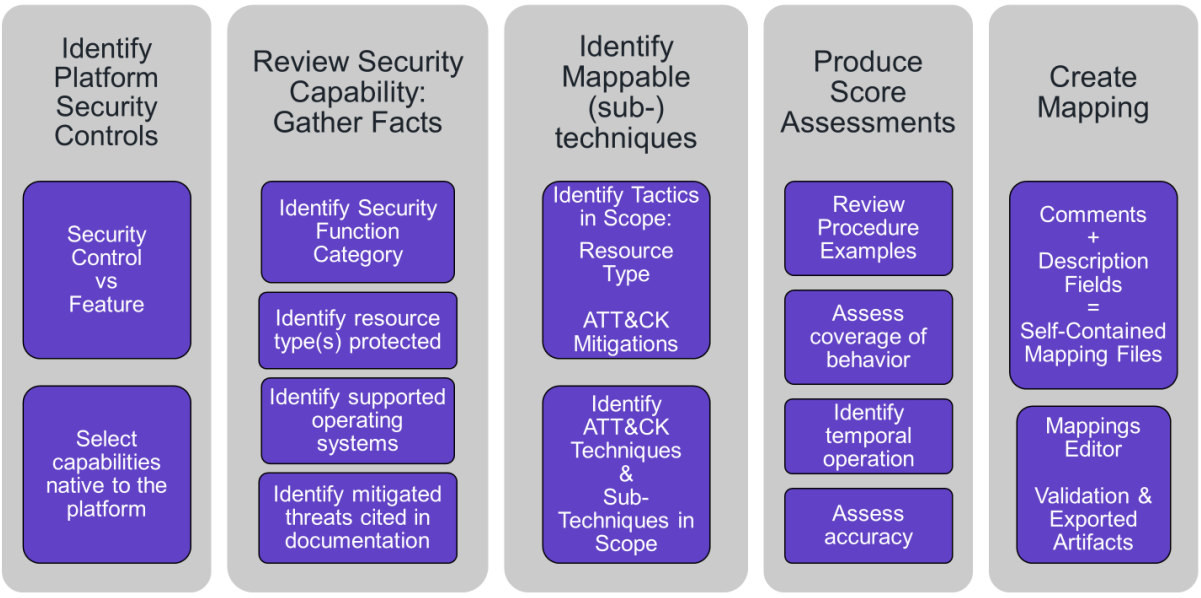
You can use Mappings Explorer for many different purposes. In this document, we will focus on using the mappings to align cyber defenses to threats by identifying security capabilities mapped to detect, defend against, or respond to specific technology platform–based branches of our attack trees. Later

in this section, we will use these resources to visualize and assess defensive coverage to identify deficiencies and plan policy and security capability implementation around the adversary TTPs from Question 2.

Creating Security Capability Mappings

The Center uses a standard methodology to map security controls native to a technology platform to ATT&CK. As discussed previously, many of these mappings have already been done for you and are readily accessible in Mappings Explorer. If you have a technology platform that has not been mapped, the below steps are a reusable method of using ATT&CK to determine the capabilities of a platform's security offerings. The methodology consists of the following basic steps:

1. Identify Platform Security Controls
   Identify the native security controls available on the platform.
2. Review Security Capability
   For each identified control, understand the security capabilities it provides.
3. Identify Mappable ATT&CK Techniques & Sub-techniques
   Identify the ATT&CK techniques and sub-techniques mappable to the control.
4. Assess and Score Control Effectiveness
   Assess the effectiveness of the type of protection the control provides (protect, detect, or response) for the identified ATT&CK techniques and sub-techniques.
5. Create a Mapping
   Create a mapping based on the information gathered from the previous steps.



Mapping Methodology

The full mapping methodology and scoring rubric are available on the Mappings Explorer website.

Creating Custom Mappings

For most users, you should start with Mappings Explorer to find mapping data relevant to your environment, which is available for downloading in spreadsheet or machine-readable formats. If you need to produce your own customized mappings data, then you can apply the mapping methodology to the platform capabilities you have.

If you are not using one of the mapping frameworks in the Mappings Explorer collection and instead plan on creating a custom mapping for your technology platform, we recommend using the Center's Mappings Editor tool and following the documentation to create new mappings.

Mappings Editor

Mappings Editor[21] is an interactive web-based tool created by the Center for creating and updating mappings of security capabilities to ATT&CK. At the time of publication, this tool is available as a public beta.

Mappings Editor makes it quick and easy to create, edit, and review mappings, and it includes several features specially engineered to enhance the mapping process. The Editor is designed to streamline the creation of mapping files, which consist of one or more mappings that associate a security control, vulnerability, or capability to an adversary behavior catalogued by ATT&CK. Using the Mappings Editor, the mapping files can be exported as ATT&CK Navigator layers or as .CSV, .JSON, .YAML, or Microsoft Excel (.XLSX) files. To get started, review the editor documentation to learn how to create the initial mappings file, and then use the link provided to launch the Mappings Editor web application.

Mitigating Threats to AMPS

For the AMPS device scenario, we will be looking at the security capabilities native to the Azure cloud platform. Using Mappings Explorer, we can easily identify 48 Azure security capabilities[22] mapped to ATT&CK techniques and sub-techniques, with a total of 978 mappings. Analyst attention can be focused on considering the applicability of these mapped security capabilities as mitigation options for the specific threats identified in Question 2.

Azure security capability mappings fall under Security Stack Mappings, which include scoring assessments for each control's ability to protect against, detect, and respond to TTPs. These assessments are provided to reflect the security capability's functions and ability to mitigate the mapped threats. Azure mappings are provided for the following capability function areas:

- Protect: capability limits or contains the impact of a (sub-)technique.
- Detect: capability identifies the potential occurrence of a (sub-)technique.
- Respond: capability provides actions to take for detected (sub-)technique.

[21] The Center for Threat-Informed Defense (2024), *Mappings Editor*: https://github.com/center-for-threat-informed-defense/mappings-editor

[22] Source: Mappings Explorer Azure mappings data, Azure v06292021 to ATT&CKv8.2 (https://center-for-threat-informed-defense.github.io/mappings-explorer/external/azure/)

Typically, it is recommended that capability mappings scored as Partial or Significant effectiveness at mitigating the behavior described by a (sub-) technique be considered for implementation. If you are inclined to include a capability scored as Minimal effectiveness, carefully consider whether this control would actually be a practical means of mitigating the threat. Often, minimally scored controls could technically mitigate the behavior, but in the real world they would not be used for that purpose. In that case, the recommendation would be to exclude them.

Using Mappings Explorer data and looking at each of the specific TTPs identified in Question 2, we identify the Azure security capabilities mappings as listed in the table below. Native Azure capabilities scored as significant or partial effectiveness for protecting against, detecting, or responding to the TTP are included, resulting in a total of 83 mappings. Note: The TTPs with strike-throughs are ones we did not score in Question 2 due to time limitations, but these would typically be used too.

Identified ATT&CK TTPs Mapped to Mitigating Azure Security Capabilities

ATT&CK (Sub-)Technique	ATT&CK ID	Mapping Category	Effectiveness Score	Azure Security Capability
Account Discovery	T1087	detect	partial	Alerts for Windows Machines
Account Manipulation	T1098	protect	partial	Azure AD Privileged Identity Management
		protect	partial	Role Based Access Control
		detect	partial	Microsoft Defender for Identity
Active Scanning	T1595	protect	partial	Azure Firewall
		protect	partial	Azure Web Application Firewall
Additional Cloud Credentials	T1098.001	protect	significant	Azure AD Privileged Identity Management
		protect	partial	Role Based Access Control
Automated Collection	T1119	protect	partial	Cloud App Security Policies
		detect	partial	Cloud App Security Policies
Brute Force	T1110	protect	significant	Azure AD Multi-Factor Authentication
		protect	significant	Conditional Access
		protect	significant	Just-in-Time VM Access
		protect	significant	Passwordless Authentication
		protect	partial	Azure Active Directory Password Protection
		protect	partial	Azure AD Identity Secure Score
		protect	partial	Azure AD Password Policy
		protect	partial	Azure Policy
		detect	significant	Azure Alerts for Network Layer
		detect	partial	Alerts for Windows Machines
		detect	partial	Azure Sentinel
		detect	partial	Cloud App Security Policies
		detect	partial	Linux auditd alerts and Log Analytics agent integration
		detect	partial	Microsoft Defender for Identity
Cloud Service Discovery	T1526	protect	partial	Azure Policy
		detect	partial	Azure Defender for Resource Manager
		detect	partial	Cloud App Security Policies
~~Create Account~~	~~T1136~~	~~detect~~	~~partial~~	~~Azure Sentinel~~
Data from Cloud Storage	T1530	protect	partial	Azure Policy
		protect	partial	Role Based Access Control
		detect	significant	Azure Defender for Storage
		detect	partial	Cloud App Security Policies
~~Exploit Public-Facing Application~~	~~T1190~~	~~protect~~	~~significant~~	~~Azure Web Application Firewall~~
		~~protect~~	~~partial~~	~~Azure Automation Update Management~~
		~~protect~~	~~partial~~	~~Azure Defender for Kubernetes~~
		~~protect~~	~~partial~~	~~Azure Policy~~

		~~protect~~	~~partial~~	~~Integrated Vulnerability Scanner Powered by Qualys~~
		~~detect~~	~~significant~~	~~Azure Web Application Firewall~~
		~~detect~~	~~partial~~	~~Alerts for Windows Machines~~
		~~detect~~	~~partial~~	~~Azure Defender for App Service~~
		~~detect~~	~~partial~~	~~Azure Network Traffic Analytics~~
Exploitation for Credential Access	T1212	protect	significant	Azure Automation Update Management
		protect	partial	Integrated Vulnerability Scanner Powered by Qualys
		detect	partial	Alerts for Windows Machines
		detect	partial	Azure Defender for App Service
File and Directory Permissions Modification	T1222	detect	partial	File Integrity Monitoring
Gather Victim Network Information	T1590	protect	partial	Azure Firewall
		protect	partial	Azure Policy
Modify Authentication Process	T1556	detect	partial	File Integrity Monitoring
Password Spraying	T1110.003	respond	significant	Azure AD Identity Protection
		protect	significant	Azure AD Multi-Factor Authentication
		protect	significant	Conditional Access
		protect	significant	Just-in-Time VM Access
		protect	significant	Passwordless Authentication
		protect	partial	Azure Active Directory Password Protection
		protect	partial	Azure AD Identity Secure Score
		protect	partial	Azure Policy
		detect	significant	Alerts for Windows Machines
		detect	significant	Azure Alerts for Network Layer
		detect	significant	Microsoft Defender for Identity
		detect	partial	Azure AD Identity Protection
		detect	partial	Azure Sentinel
		detect	partial	Cloud App Security Policies
		detect	partial	Linux auditd alerts and Log Analytics agent integration
Remote System Discovery	T1018	protect	partial	Azure Firewall
Steal Application Access Token	T1528	protect	partial	Azure AD Identity Secure Score
		protect	partial	Azure Key Vault
		protect	partial	Cloud App Security Policies
		protect	partial	Role Based Access Control
		detect	partial	Cloud App Security Policies
Unsecured Credentials	T1522	protect	partial	Azure Key Vault
Unused/Unsupported Cloud Regions	T1535	protect	partial	Azure Policy
		detect	partial	Cloud App Security Policies
Valid Accounts	T1078	respond	partial	Azure AD Identity Protection

		detect	partial	Alerts for Windows Machines
		detect	partial	Azure AD Identity Protection
		detect	partial	Azure Sentinel
		detect	partial	Cloud App Security Policies
Vulnerability Scanning	T1595.002	protect	partial	Azure Firewall
		protect	partial	Azure Web Application Firewall
		detect	partial	Azure Defender for App Service
		detect	partial	Azure Sentinel
		detect	partial	Azure Web Application Firewall

The next table presents the Azure Security Capability mappings that can provide mitigation for the ATT&CK TTPs identified in Question 2. The included capabilities were scored as being significant or partial effectiveness for each of the mapping categories of protect, detect, and respond in relation to the mapped technique.

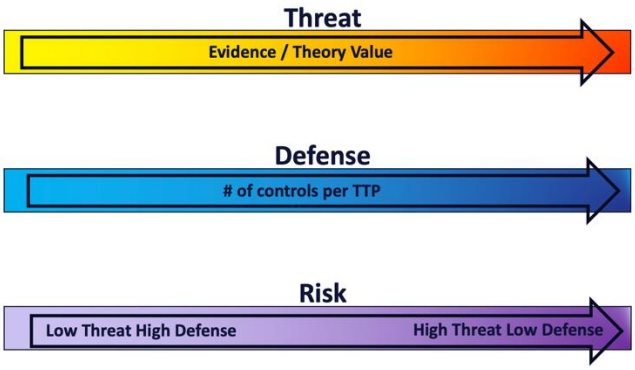
Azure Security Capability Mitigation of Identified ATT&CK TTPs

Azure Security Capability	Mapping Category	Effectiveness Score	ATT&CK ID	ATT&CK (Sub-)Technique
Alerts for Windows Machines	detect	significant	T1110.003	Password Spraying
	detect	partial	T1087	Account Discovery
	detect	partial	T1110	Brute Force
	~~detect~~	~~partial~~	~~T1190~~	~~Exploit Public-Facing Application~~
	detect	partial	T1212	Exploitation for Credential Access
	detect	partial	T1078	Valid Accounts
Azure Active Directory Password Protection	protect	partial	T1110	Brute Force
	protect	partial	T1110.003	Password Spraying
Azure AD Identity Protection	respond	significant	T1110.003	Password Spraying
	respond	partial	T1078	Valid Accounts
	detect	partial	T1110.003	Password Spraying
	detect	partial	T1078	Valid Accounts
Azure AD Identity Secure Score	protect	partial	T1110	Brute Force
	protect	partial	T1110.003	Password Spraying
	protect	partial	T1528	Steal Application Access Token
Azure AD Multi-Factor Authentication	protect	significant	T1110	Brute Force
	protect	significant	T1110.003	Password Spraying
Azure AD Password Policy	protect	partial	T1110	Brute Force
Azure AD Privileged Identity Management	protect	significant	T1098.001	Additional Cloud Credentials
	protect	partial	T1098	Account Manipulation
Azure Alerts for Network Layer	detect	significant	T1110	Brute Force
	detect	significant	T1110.003	Password Spraying

Azure Automation Update Management	protect	significant	T1212	Exploitation for Credential Access
	~~protect~~	~~partial~~	~~T1190~~	~~Exploit Public-Facing Application~~
Azure Defender for App Service	~~detect~~	~~partial~~	~~T1190~~	~~Exploit Public-Facing Application~~
	detect	partial	T1212	Exploitation for Credential Access
	detect	partial	T1595.002	Vulnerability Scanning
Azure Defender for Kubernetes	~~protect~~	~~partial~~	~~T1190~~	~~Exploit Public-Facing Application~~
Azure Defender for Resource Manager	detect	partial	T1526	Cloud Service Discovery
Azure Defender for Storage	detect	significant	T1530	Data from Cloud Storage
Azure Firewall	protect	partial	T1595	Active Scanning
	protect	partial	T1590	Gather Victim Network Information
	protect	partial	T1018	Remote System Discovery
	protect	partial	T1595.002	Vulnerability Scanning
Azure Key Vault	protect	partial	T1528	Steal Application Access Token
	protect	partial	T1522	Unsecured Credentials
Azure Network Traffic Analytics	~~detect~~	~~partial~~	~~T1190~~	~~Exploit Public-Facing Application~~
Azure Policy	protect	partial	T1110	Brute Force
	protect	partial	T1526	Cloud Service Discovery
	protect	partial	T1530	Data from Cloud Storage
	~~protect~~	~~partial~~	~~T1190~~	~~Exploit Public-Facing Application~~
	protect	partial	T1590	Gather Victim Network Information
	protect	partial	T1110.003	Password Spraying
	protect	partial	T1535	Unused/Unsupported Cloud Regions
Azure Sentinel	detect	partial	T1110	Brute Force
	~~detect~~	~~partial~~	~~T1136~~	~~Create Account~~
	detect	partial	T1110.003	Password Spraying
	detect	partial	T1078	Valid Accounts
	detect	partial	T1595.002	Vulnerability Scanning
Azure Web Application Firewall	~~protect~~	~~significant~~	~~T1190~~	~~Exploit Public-Facing Application~~
	protect	partial	T1595	Active Scanning
	protect	partial	T1595.002	Vulnerability Scanning
	~~detect~~	~~significant~~	~~T1190~~	~~Exploit Public-Facing Application~~
	detect	partial	T1595.002	Vulnerability Scanning
Cloud App Security Policies	protect	partial	T1119	Automated Collection

	protect	partial	T1528	Steal Application Access Token
	detect	partial	T1119	Automated Collection
	detect	partial	T1110	Brute Force
	detect	partial	T1526	Cloud Service Discovery
	detect	partial	T1530	Data from Cloud Storage
	detect	partial	T1110.003	Password Spraying
	detect	partial	T1528	Steal Application Access Token
	detect	partial	T1535	Unused/Unsupported Cloud Regions
	detect	partial	T1078	Valid Accounts
Conditional Access	protect	significant	T1110	Brute Force
	protect	significant	T1110.003	Password Spraying
File Integrity Monitoring	detect	partial	T1222	File and Directory Permissions Modification
	detect	partial	T1556	Modify Authentication Process
Integrated Vulnerability Scanner Powered by Qualys	~~protect~~	~~partial~~	~~T1190~~	~~Exploit Public-Facing Application~~
	protect	partial	T1212	Exploitation for Credential Access
Just-in-Time VM Access	protect	significant	T1110	Brute Force
	protect	significant	T1110.003	Password Spraying
Linux auditd alerts and Log Analytics agent integration	detect	partial	T1110	Brute Force
	detect	partial	T1110.003	Password Spraying
Microsoft Defender for Identity	detect	significant	T1110.003	Password Spraying
	detect	partial	T1098	Account Manipulation
	detect	partial	T1110	Brute Force
Passwordless Authentication	protect	significant	T1110	Brute Force
	protect	significant	T1110.003	Password Spraying
Role Based Access Control	protect	partial	T1098	Account Manipulation
	protect	partial	T1098.001	Additional Cloud Credentials
	protect	partial	T1530	Data from Cloud Storage
	protect	partial	T1528	Steal Application Access Token

Identify Areas of Risk



During this step of the process, we will be combining scored threat TTPs that were compiled from the Evidence and Theory sections with the defensive capabilities mapped in the previous section. The example will continue to focus on the Azure platform and the TTPs associated with possible threats against the AMPS device. This step results in three Navigator Layers; the layers are optional and can be completed or not based on the organization's needs.

Start by creating two Navigator Layers and overlaying them for a comprehensive view:

Layer 1: A visualization of the threat scoring determined in Question 2 (see figure below). To create this layer within Navigator, the following numbering will be used:

5 = No theory, No evidence

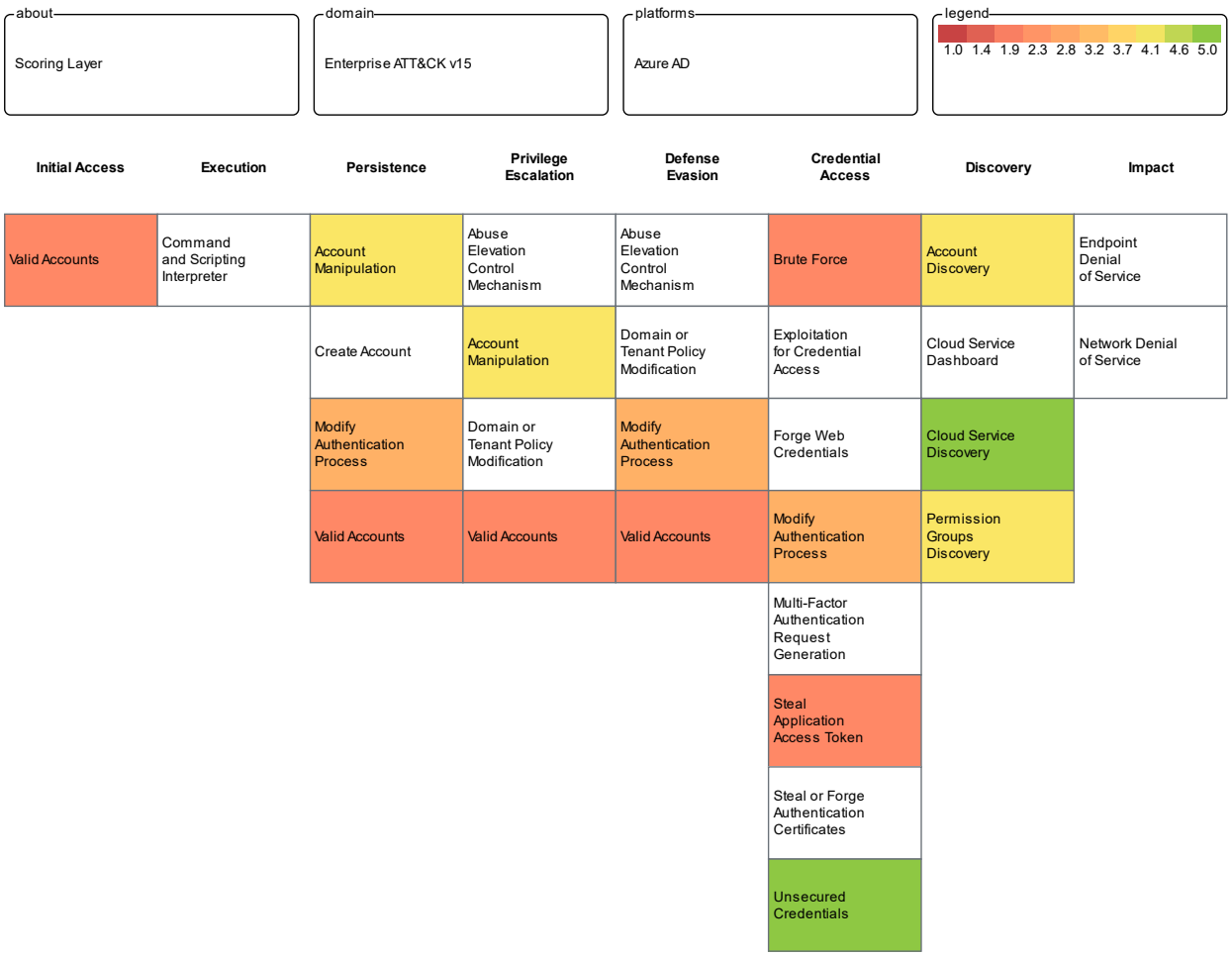
4 = No theory, Some evidence or Some theory, No evidence

3 = No Theory, Strong Evidence or Some theory, Some Evidence or Strong theory, No evidence

2 = Some theory, Strong evidence or Strong theory, Some evidence

1 = Strong theory, Strong evidence

Example: T1556: Modify Authentication Process = Some theory Some Evidence = 3

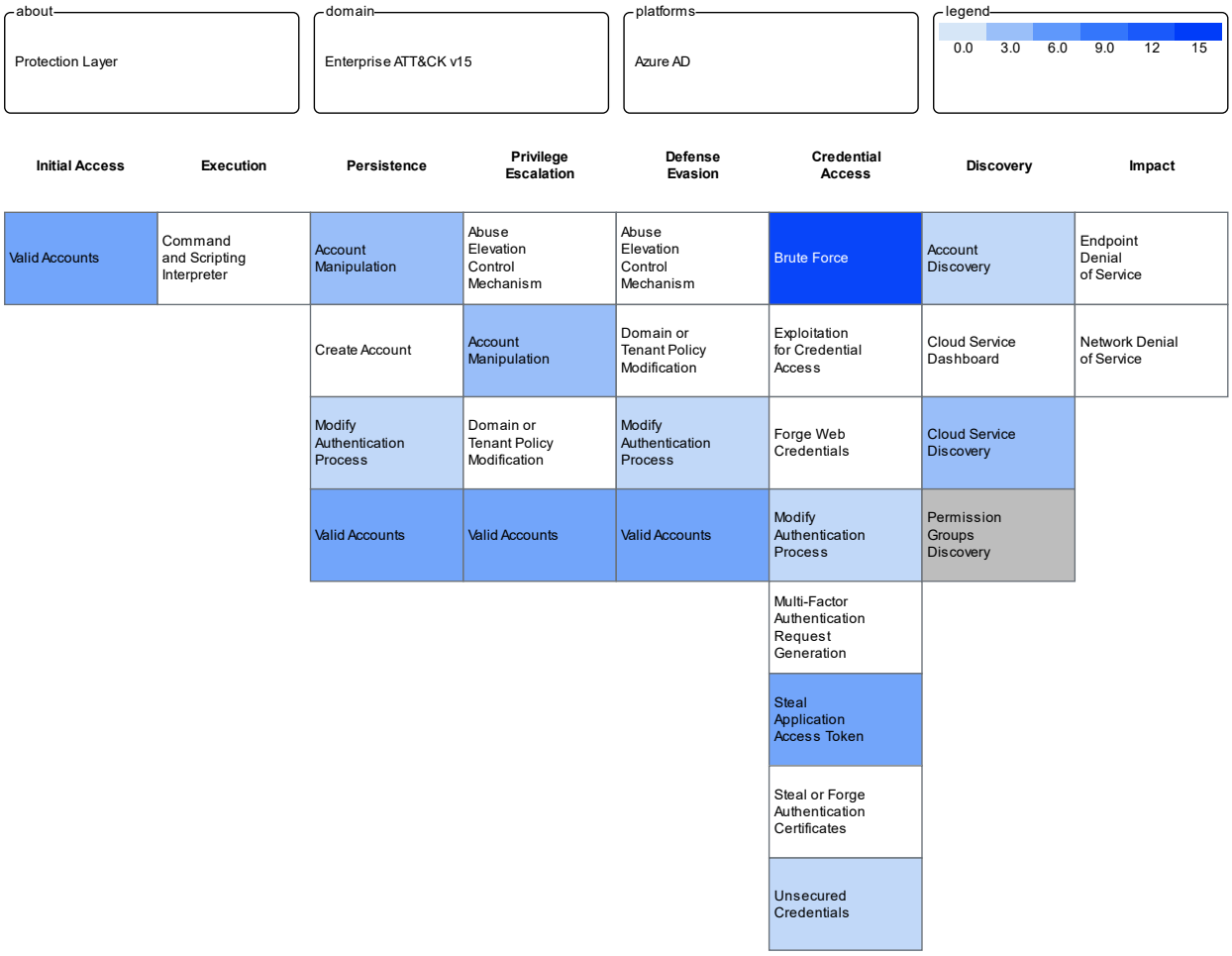


Example ATT&CK Navigator Layer for Scored TTPs

Layer 2: A visualization of the number of defensive controls determined in the Question 3 mappings (see figure below). To figure out this range, you will count the amount of defensive capabilities for each TTP, take the highest amount, and make that the maximum, with the minimum being 1.

```
T1556: Modify Authentication Process # of defensive capabilities = 1

Maximum # of defensive capabilities = 15 (Password Spraying)
```

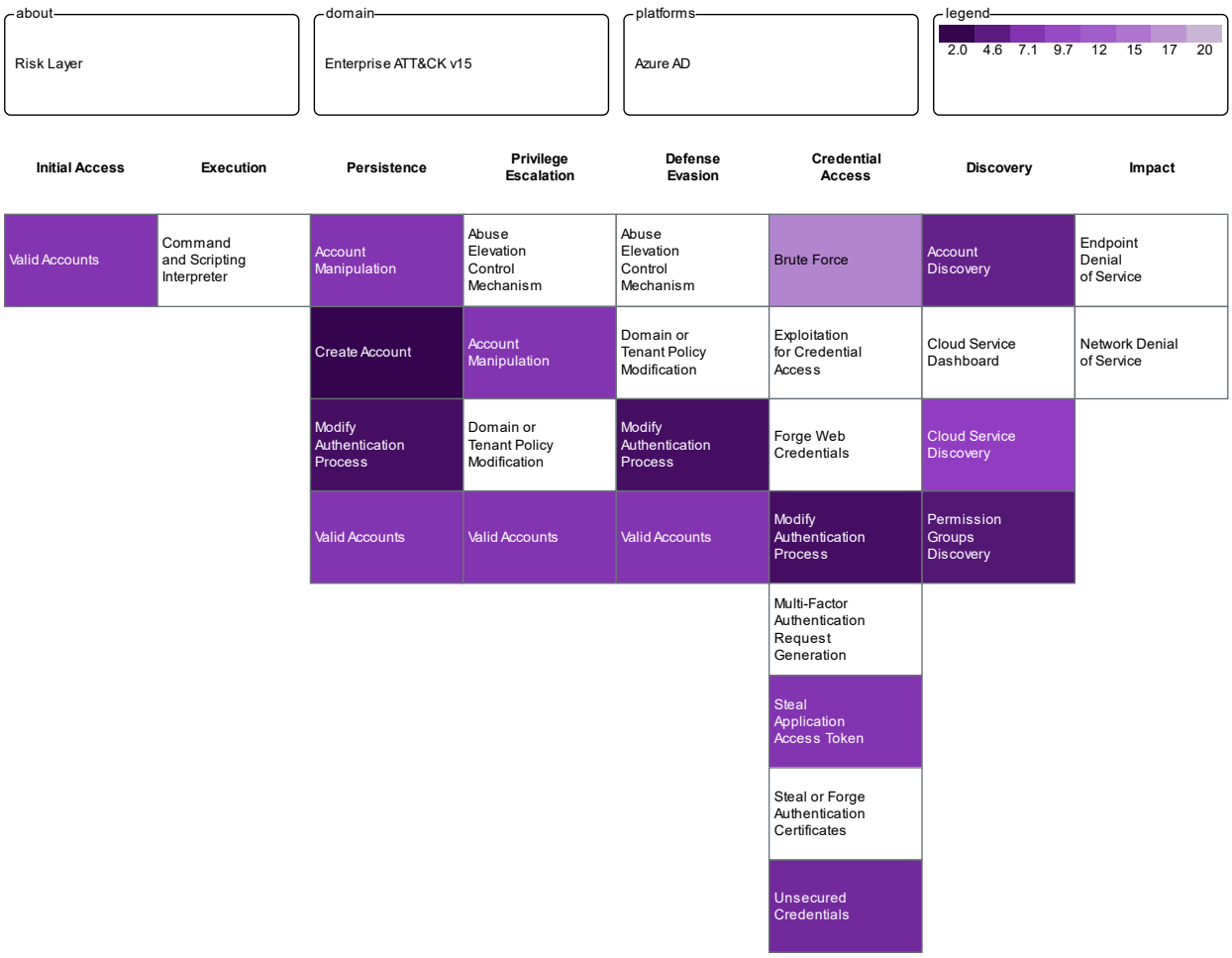


Example ATT&CK Navigator Layer for Number of Defensive Capabilities

Once those two layers are completed, you overlay them to create a heat map that visualizes the overall risk. On the low end we have low threat high defense and on the high end we have high threat low defense. An easy way to determine this is by adding the maximum determined for Layer 2 (in our case 15) to the maximum for Layer 1 (which should always be 5). The resulting number will determine the range to set for the Navigator gradient (in our case 15 + 5 = 20). Then, for each TTP, the associated numbers for Layer 1 and Layer 2 will be combined. See below example risk scoring for T1556 Modify Authentication Process.

```
T1556 Modify Authentication Process:
      Some theory, Some Evidence = 3
      # of defensive capabilities = 1
      Navigator value: 4
      Navigator scale: 2 – 20
```

When these are plotted on the Navigator Layer, light purple is low risk and dark purple is high risk.



Example ATT&CK Navigator Layer for Risk (Scoring + Defensive Capabilities)

Implementing Mitigations to Risks

At this stage, by leveraging the Mapping Explorer or crafting mappings of our own, we understand the mitigations within our environment and the degree to which each addresses the threats we are likely to face. By implementing these specific Azure controls that we've mapped to our relevant threat TTPs, we've significantly reduced the potential impact of an attack.

By reviewing our overlayed Navigator Layers, we can see that several TTPs, such as "Valid Accounts" (T1078), remain a high risk to our system even with existing mitigations implemented within our Azure environment. Addressing these latent risks is a priority, and your team may already have applicable controls that address these risks if put in place. If not, MITRE ATT&CK can serve as a resource for finding additional mitigations addressing these risks. ATT&CK directly links each TTP with associated mitigations that can be leveraged to similarly address the risk posed by that TTP. Currently, there are seven mitigations tied to T1078. For instance, Account Use Policies (M1036) recommends using conditional access policies to control for log-ins from insecure or unknown devices.

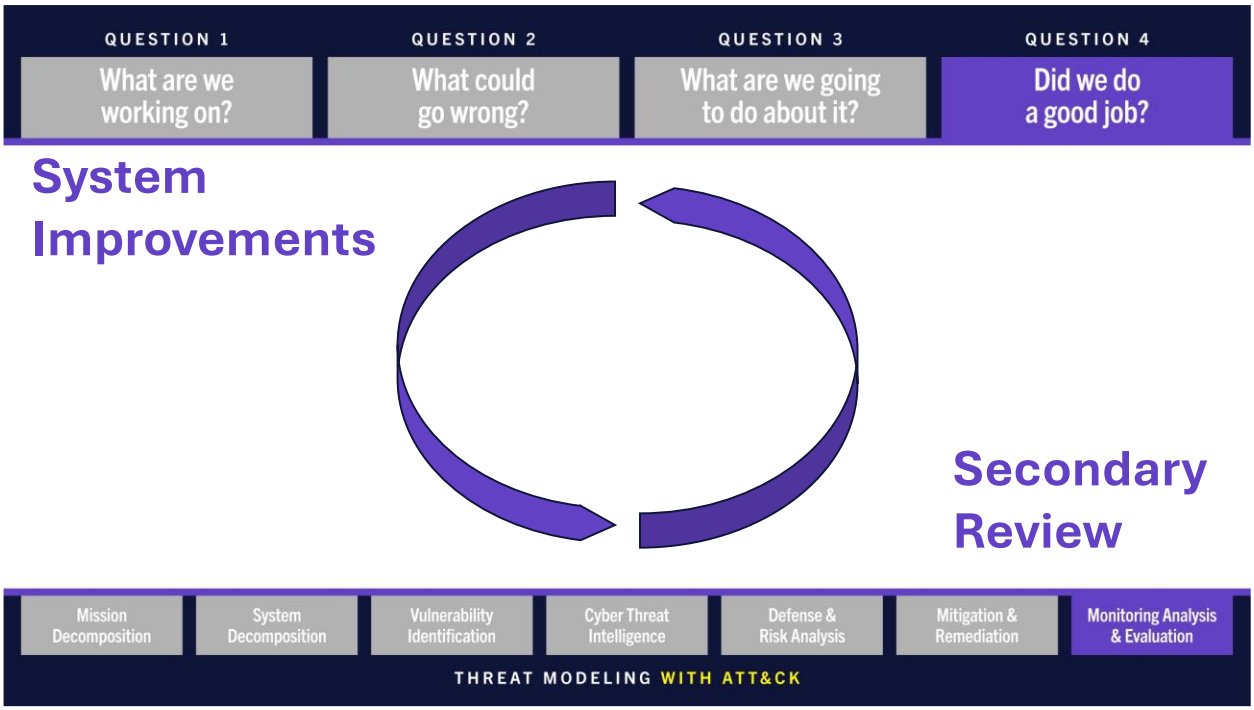
In addition to the mitigation details within ATT&CK, Mappings Explorer provides ATT&CK TTPs mapped to NIST 800-53. NIST 800-53 is a catalog of security and privacy controls for managing cybersecurity and privacy risks to information systems that, if tailored and implemented for your environment, can help mitigate the latent risk posed by our remaining threats. NIST SP 800-53 controls can be identified by their Capability IDs, which consist of a two-letter identifier for each control family (e.g., AC for Access Control), followed by a dash and two numbers for each base control (e.g., AC-02).("XX-##").

The Valid Account technique T1078, for example, is mapped to several NIST 800-53 controls. These include AC-02 Account Management, AC-03 Access Enforcement, and AC-06 Least Privilege. These controls provide safeguarding measures that can be adopted within your system to better protect against your remaining high-risk TTPs. In our case, one mitigation might be prohibiting or restricting the use of system services or software to achieve "least functionality." This can be done by ensuring component functionality is limited to a single function per component, removing unused or unnecessary software, or limiting unnecessary physical and logical ports and protocols to prevent unauthorized connection of components, transfer of information, and tunneling. These mitigations can be tailored further to fit your given system by collaborating with your team on potential implementations.

These mappings provide an important resource for assessing security control coverage against real-world threats, and they provide a foundation for integrating ATT&CK-based threat information into the risk management process. This allows users to focus their time and resources on understanding how controls map to threats in their specific environment and to implement additional protections tailored to specific system risks. Implementation of the mapped controls should be done in whatever manner satisfies system and organizational security or business needs. Tailored security capabilities and the selection of appropriate controls for our system and environment are essential for securing systems against potential threats.

Implementing one security control can improve mitigations for multiple threats. When considering which controls to implement for your system, check the effect on the other threat risk ratings of concern. For example, employing the principles of AC-06 Least Privilege, allowing only authorized accesses for users that are necessary to accomplish assigned organizational tasks can help mitigate more than 250 ATT&CK TTPs.

Question 4: Did we do a good job?



As you continue with your system's development or sustainment process, threat model in hand, your team can make use of a variety of approaches to evaluate the success of your Q3 mitigations.

System Improvements

The first approach reflects the degree to which your threat model has informed the development of your system.

- For systems still in development, identify design decisions influenced by your threat modeling analysis.
- For systems already deployed, identify actionable outcomes where changes to your infrastructure may take place due to your threat modeling analysis.

Alternatively, your team may call for a security assessment, in which an internal or external team could evaluate or probe your system to determine its security and whether the controls you've deployed across your system are effective.

While these sources of feedback, and others, can be drawn upon with varying degrees of complexity, the most effective means of evaluating your mitigations is with a secondary review.

Secondary Review

When performing periodic reevaluations, your team should ask key questions and review associated metrics to ensure existing implemented controls are reviewed and, if needed, updated to maintain effectiveness and currency with organizational objectives.

The purpose of a secondary review is to effectively reassess your threat model, determine remaining risks, and figure out what additional defensive actions need to be taken. Some valuable questions include:

- Are your existing risk ratings correct? Should they be changed given new theoretical or evidence-based findings?
- Does your team have the right composition? Are you looping in stakeholders with a diverse range of backgrounds and perspectives?
- What additional changes have been made to your system since the last review? Does your existing model accurately reflect their state of deployment?
- Are the same critical assets being used to accomplish the system's purpose? Have certain security controls become obsolete or redundant?

There are existing processes or data sources you can leverage to answer these questions. Perhaps your organization has a process for system risk acceptance, or you actively track system patches and compliance metrics. These can all inform your secondary review and give you the answers you need. From this secondary review, you'll be able to ensure that your mitigations are sufficiently tailored to your system as it evolves with time.

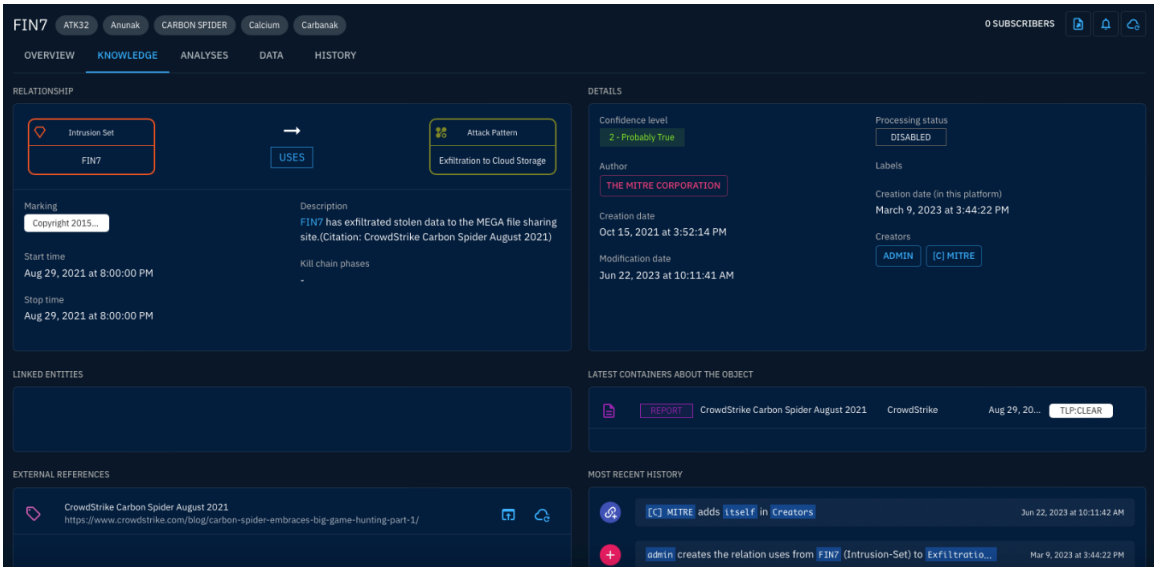
Appendix

Cyber Threat Intelligence Resources

Leveraging existing CTI allows you to develop known attack vectors that could be used against your system. There are many resources for CTI data and this appendix is made to refence a few that we have found useful.

- The Center's Sightings Ecosystem (https://mitre-engenuity.org/cybersecurity/center-for-threat-informed-defense/our-work/sightings-ecosystem/) project is an example of data that can be leveraged throughout this process to help identify, or highlight, commonly seen TTPs. At the time of publish, their work consists of over 1.6 million sightings of 353 unique techniques from almost 200 countries.
- Many venders publish opensource reports on blogs or their websites. Monitor these sources for new/relevant reports. Attack Flow created best practices for selecting open-source reports and this can be beneficial during this step:
- "Reports should be transparent about where the data originates and provide a technically competent overview of an incident.
- Reports should originate from a vendor with a track record of accurate reporting and first-hand analysis of the incident in question.
- Reports should provide the most current information on the malware or breach.
- Reports should make it easy to identify any information gaps. Use multiple sources to address gaps and corroborate the data, if possible.
- Reports should distinguish between facts, assumptions, and analytical assessments.
- When available, use attribution and targeting information from reports to enrich your attack flows."
- When it comes to researching CTI for embedded systems, MITRE developed a publicly available knowledge base called EMB3D. This is a great resource for both theory and evidence. Start by down selecting by embedded system property and read through the various threats to each.

It is a good idea to have a central location/repository for all your CTI data. This can be a spreadsheet or a threat intelligence platform (TIP) like OpenCTI (see example data below for FIN7). There are many TIP out there that will do to research work for you – automatically pulling in the latest vender reports. Some TIPs will even auto-parse the data in reports for you. Be sure to spot check any automated report parsing for accuracy.



Image[15]*:* FIN7 Attack Pattern CTI

[15] The MITRE Corporation. “FIN6 | Attack Pattern | Exfiltration to Cloud Storage.” *OpenCTI*, 2021