Practical Cyber Security for Cyber Security Practitioners

Prof. Sandeep Kumar Shukla

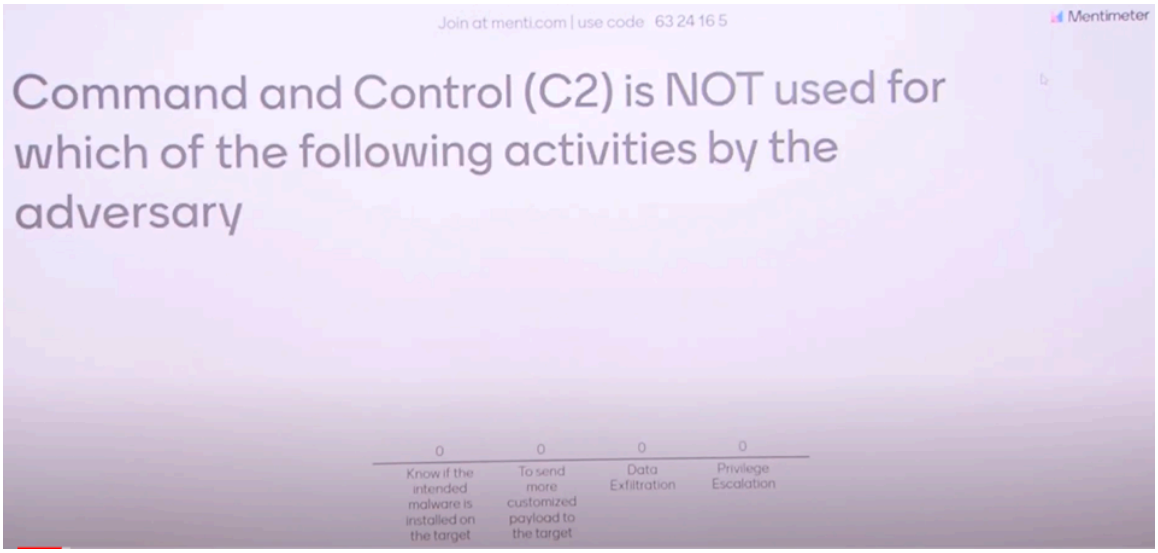
Department of Computer Science and Engineering

Indian Institute of Technology, Kanpur

Lecture 05

Introduction to MITRE ATT & CK framework

So now tell me, command and control is not used for which of the following activities by the adversary. So, I have four different activities here. Remember, it's not.

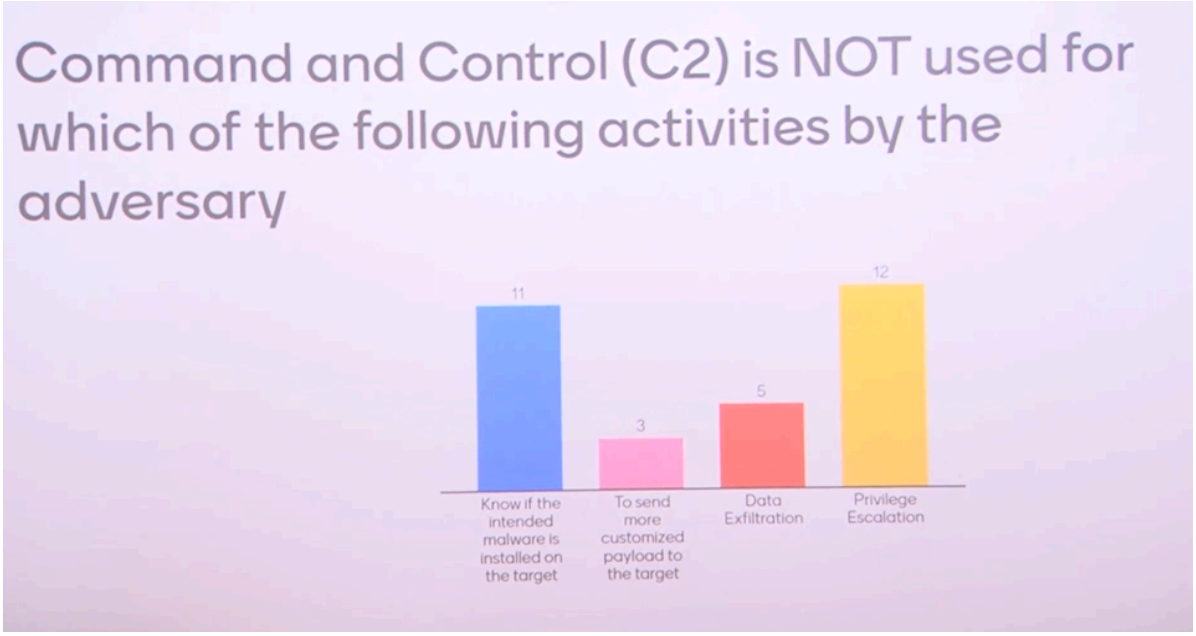


Suppose you are the one sending malware to somebody else's machine, and you want to know if the malware has been installed there. How would you do that? You have to have the malware communicate to you.

The first choice is that the command and control wants to know, the adversary wants to know if the malware has been installed. Then it will write the malware in such a way that as soon as the malware finds a target and executes, it will call on the network functions and communicate to the command and control. Isn't it? How else will the adversary know that the malware actually got installed?

Now, once the adversary knows that it has been installed, then it will want the malware to find something on that machine—what applications are running, what versions are running, what are the different files in the file system, if there are any credentials

somewhere in that machine, and if there is a weak implementation of a protocol through which it can move.



So, all this information the malware will send to the adversary via the command and control route. Then the adversary, based on the information it got, will customize a payload that can exploit the particular situation that the malware is reporting to C2. Therefore, the second one is also something that C2 is used for, right? To get a better understanding of its target and customize a more virulent payload for the target.

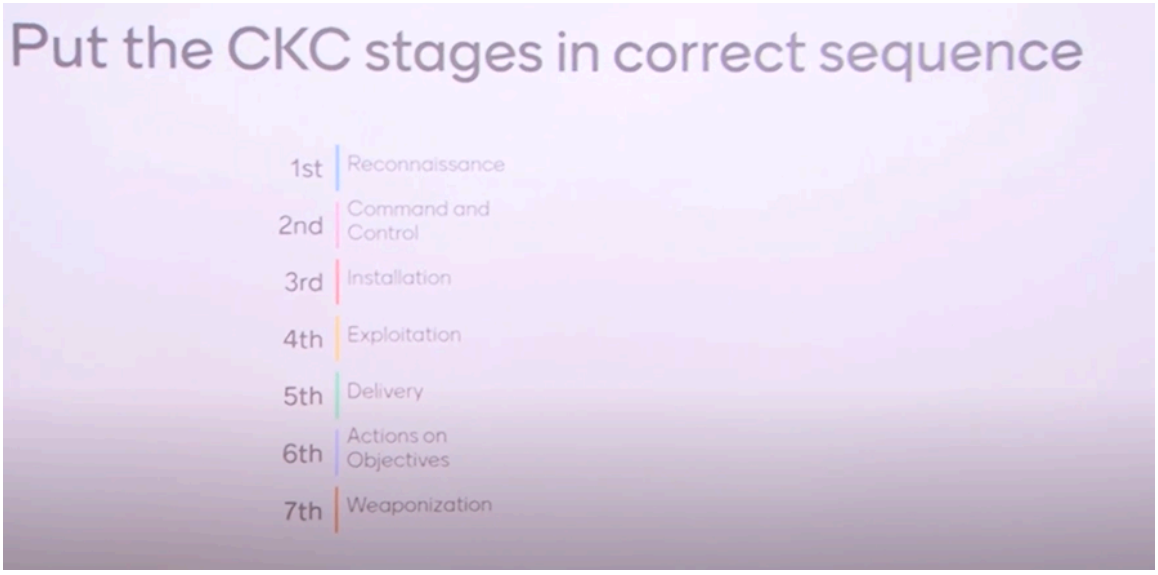
Now, if you want to do data exfiltration, let's say you want to exfiltrate data from another person's system using malware, how will that malware send the data? Where will it send the data? It will read the data from the target machine, but it has to send it somewhere.

So, that has to be the command and control server, right? So, all these three choices are not correct because I am asking which one of these is not a use of command and control. I am not asking which one is a use of command and control because that would make sense since I have three different choices, all of which are actually uses of command and control. The last one, privilege escalation, is the natural choice because privilege escalation is a very local thing. It has nothing to do with what happens on the command and control side. If there is a weak program that has a privilege escalation vulnerability, your homework will make you do a privilege escalation.

So, you will understand how privilege escalation happens. In terms of homework, you

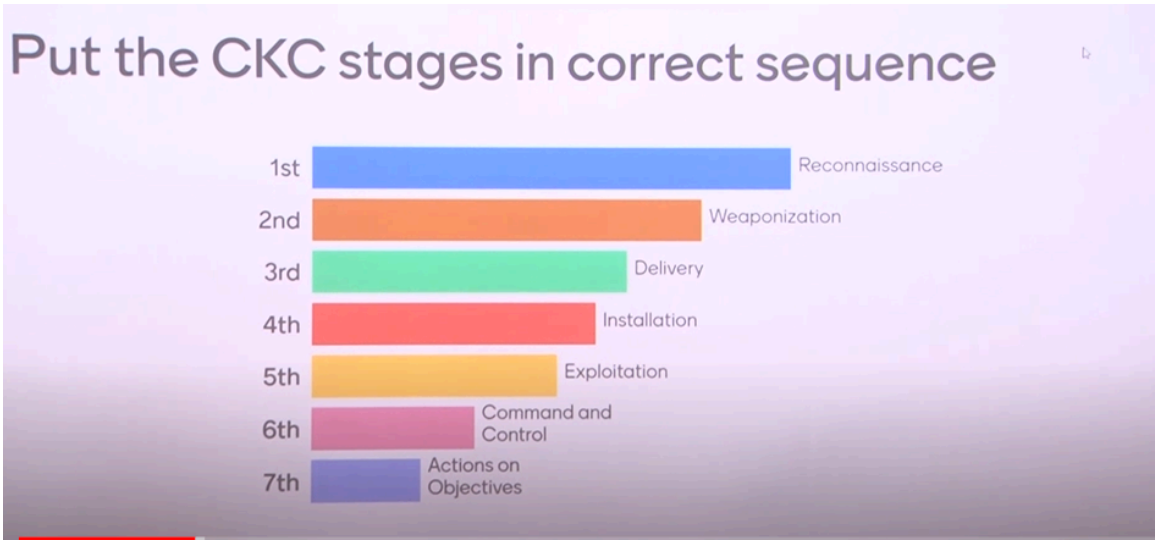
will get virtual machines that you will have to install on your machine and perform all these tasks on that virtual machine.

Okay, so next one. This is an easy one. Use your finger to push up and down and sort them in the order in which they appear in CKC.

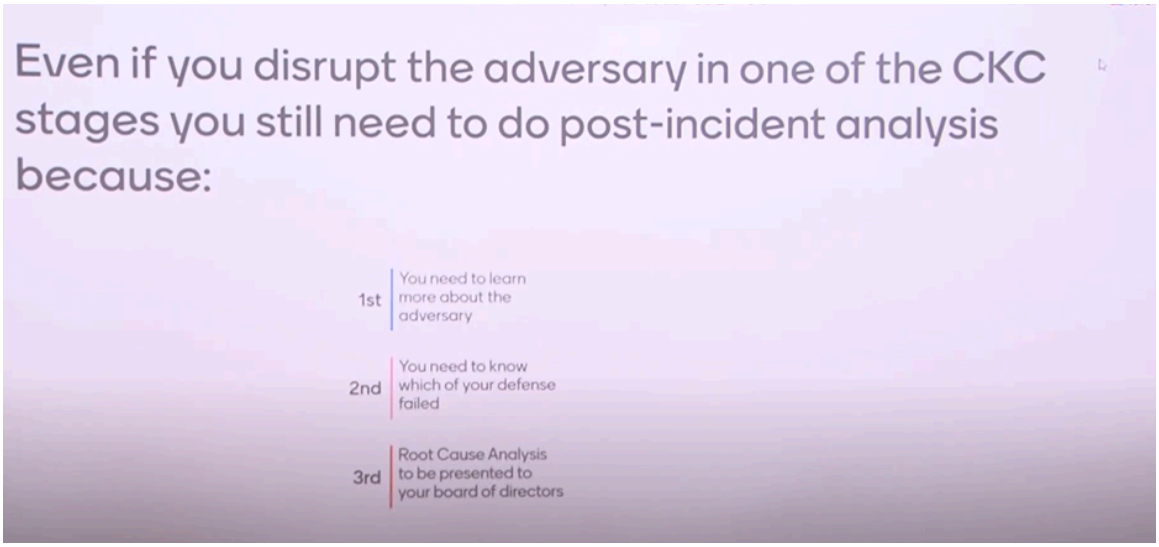


So, these are the seven stages of the cyber kill chain. They're in a random order. You have to push them up and down to put them in the right order.

I see there are not many responses yet. It is almost correct. Where is it not exactly correct? See, you have to exploit a weakness in the system before you can do installation, right? So your exploitation and installation order, for the majority, is in the opposite order. But otherwise, you've got the other ones right. This one is just a little bit in the reverse order.



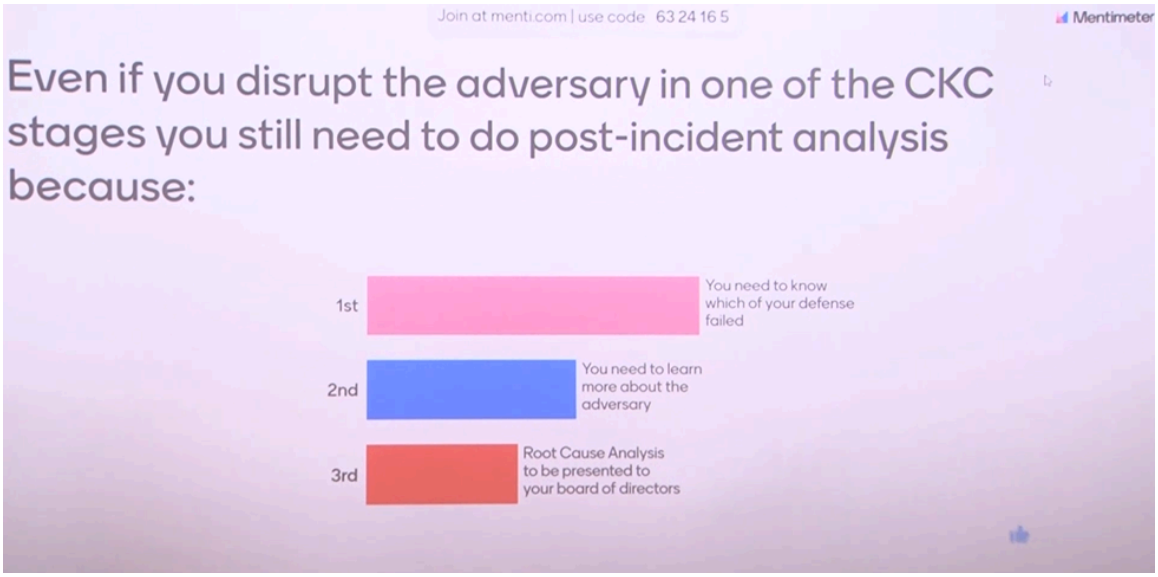
Okay, so now go to the next one.



Here, I'm asking, suppose you disrupt, remember in the CKC seven stages, the claim of CKC is that if your defense can actually stop them in one of the seven stages, then you win, right? You prevent the adversary from doing the final thing it wants to do. Now, the question here is whether you stop it or not, you have to do post-incident analysis. There are three reasons given why, and you have to say which one is the most important reason why I would like to do the post-incident analysis and not just be happy that the bad thing didn't happen. 'All's well that ends well' doesn't work here. You have to actually analyze why it could do what it could do.

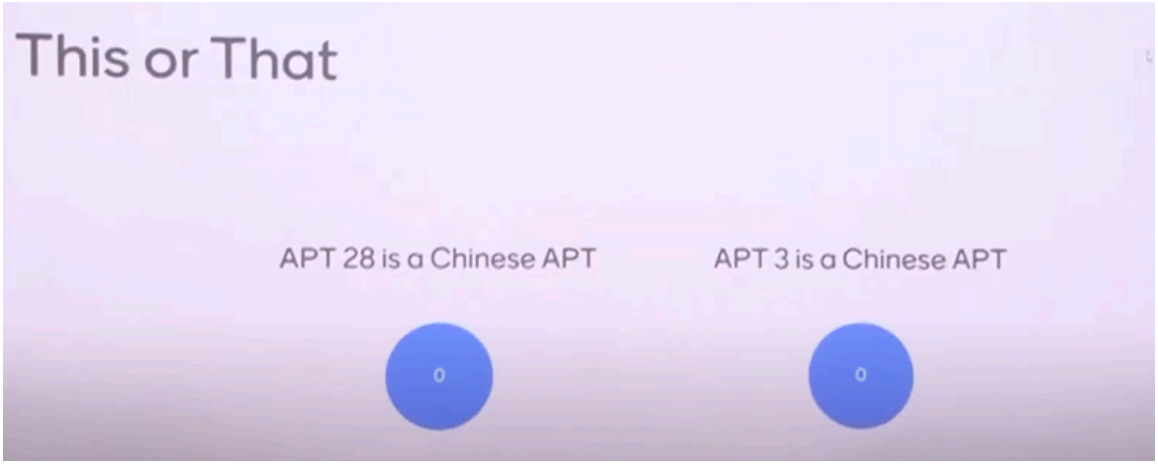
Okay, so this ordering is rather subjective. Of course, you have to know where the defense failed, right? Then you have to fix that because your defense must have failed in at least one of the stages up to whichever stage the adversary could get in. Until that

stage, your defense didn't work, at least against that particular adversary. So, you have to figure out what failed and then accordingly fix those issues.



Now, you can debate about the second and third points. Of course, you need to learn more about the adversary, but you also have to do root cause analysis. In a well-governed cybersecurity environment, every incident's root cause analysis is presented to the highest authority to inform them of the possible risks in the organization. So, you can have a second and third kind of risk condition.

Now, this one I have put intentionally. I haven't really told you about all possible APT groups, advanced persistent threat groups. I have said that advanced persistent threat groups are very resourceful threat groups, usually supported or funded by nation-state governments.

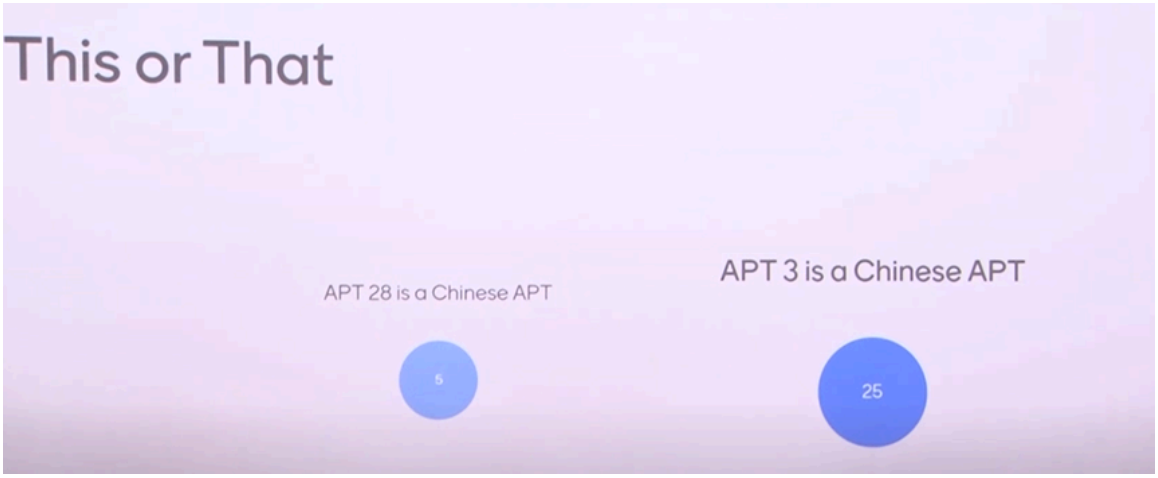


And it's actually quite difficult to tell whether a particular threat group is working for a specific government. This process is called attribution. Attribution is difficult, but there

are some which have been analyzed by a lot of threat intelligence companies, and we kind of know which ones are correct and some of them we do not know as fully correct.

So, in this case, I wanted to see if you got interested beyond the class and actually did some studies about these nation-state adversaries. In any case, APT28 is not a Chinese APT; it's actually a Russian APT. They were responsible for the SolarWinds attacks in 2020 in the US. Many US government entities and organizations were infiltrated by the supply chain attack on a software system for network monitoring called SolarWinds.

So, APT28 is not a Chinese group. Most of you have avoided that. And indeed, APT3 is a Chinese threat group.



So, most of you have looked at that, so that's good. Now, for each of these, you have to say whether it's true or false. Well, the first one I have already disclosed. So, for the other two. APT28, nobody got wrong.

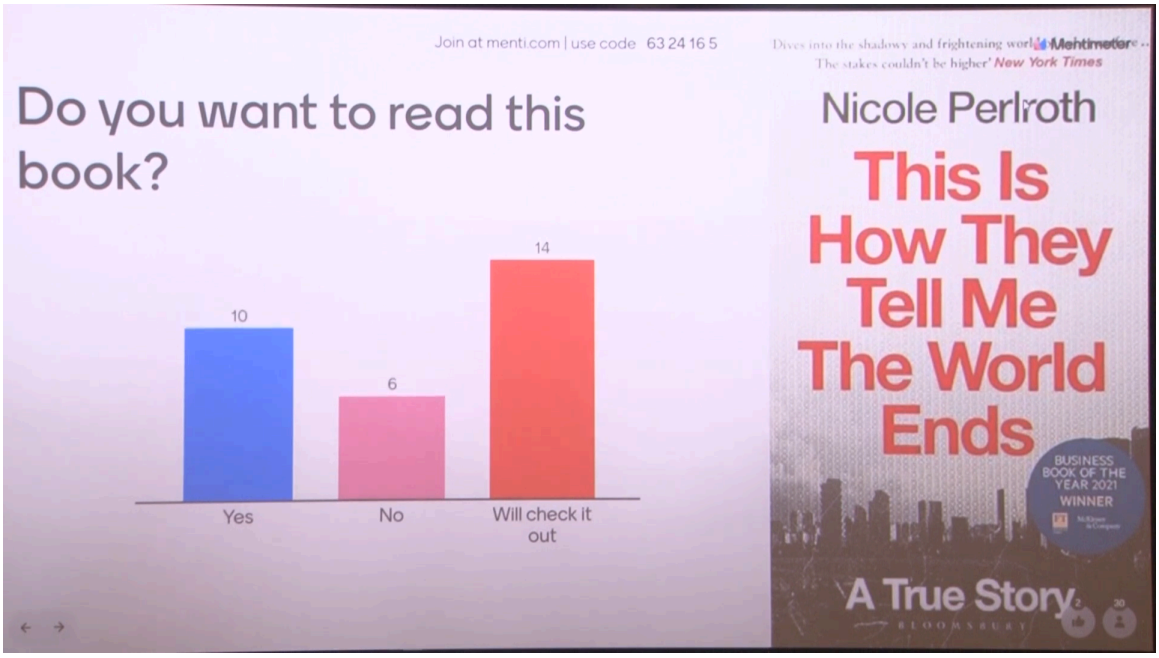
So, APT37 is indeed a North Korean group and sometimes it is considered the same as the Lazarus group. They actually go after countries like South Korea and the US. They have been found in India also. They are pretty resourceful and a very skilled set of hackers. APT33 is also correctly an Iranian group.

So, APT33 is an Iranian hacker group. As you can imagine, countries like North Korea, Russia, Iran, and China have some of the most notorious threat groups. They have multiple different threat groups, not just one.

Now, remember that when I say something like APT3 is a Chinese threat group and APT1 is also a Chinese threat group, it may be that APT1 and APT3 are the same set of people. Based on the attacks they use, the malware they use, the command and control infrastructure they use, and the kind of targets they choose, all these things allow a threat intelligence company or organization to cluster many attacks together and name them as an APT group.

Now, it may so happen that what we are calling APT28 may actually be two different groups who are all using a similar set of malware, a similar set of attack modus operandi, and so on. It can also be the case that APT1 and APT3 are the same group, using two different sets of infrastructure or different types of malware for different types of attacks. So, all these things are shrouded in mystery, right? We do not really know that APT28 is directly talking to, for example, Putin, right? We do not know that, but the threat intelligence companies over time have analyzed and found fragments of Russian language comments in their code. They found command and control infrastructure that is not necessarily in Russia but has been found to be used by Russians in other places. They also find the times of day when they were most active.

They also find targets that they choose, like Ukraine and the US; these are mostly their targets. From that, they actually came up with the idea that this is Russian. Now, in India, we do not have this capability of attribution. So, in C3i Hub, we are doing a lot of work on this attribution, but in general, we haven't developed this attribution capability so far in India.



The last question. This is a book I already mentioned. It is a book by a New York Times cybersecurity reporter. If you remember—well, you are probably too young to remember—how many of you have heard about Snowden? Snowden was a consultant employee at, I believe, Booz Allen Hamilton, which is a defense contractor. He exfiltrated a lot of data during the early 2000s about many secret programs that the US military and intelligence agencies were conducting, including spying on its own citizens.

What he did was give this information to certain news organizations, with the New York Times being one of them. Nicol Palroth worked on that team. Since then, she discovered that there are a lot of programs by governments—not necessarily only Russian, Chinese, and Ukrainian, the usual suspects.

It's not only the usual suspects. It's actually governments like the US government, our government, the European government; they all have programs to find vulnerabilities. And, of course, Israel finds vulnerabilities in very highly used software systems, right? For example, in iOS, Android, Windows, or Windows Office—things that are widely used. Governments buy these vulnerabilities from hackers who are black hat hackers, who are not necessarily considered responsible hackers. Responsible hackers, when they find a vulnerability, do what we call a responsible disclosure. They go and tell the

company, 'Look, you have this problem. I'm going to publish this in the Black Hat conference or whatever conference, but I will wait until you fix it.

So, that's what responsible disclosure is. They won't disclose it to the world until it is fixed. Unfortunately, black hat hackers are the opposite. They find vulnerabilities but do not disclose them to the organization responsible for the software, hardware, etc. Instead, they sell them on the black market.

One of the biggest buyers in this black market is governments. Governments actually buy these exploits, for example, the National Security Agency in the US, and then they use them, right? They use them against other countries, targeting important personnel like prime ministers or other significant figures. Now, there are companies that also create these exploits and develop complete command and control systems. You can buy the entire command and control system from them.

One of the famous companies you might have heard of is NSO. NSO is the company responsible for Pegasus. Pegasus was a zero-click and zero-day malware. They sell this whole command and control infrastructure to governments, allowing them to see what is happening on someone else's phone and spy on them. They can spy on them and even plant incriminating evidence on their phone or desktop, which can later be used against them.

There is a whole business around these vulnerabilities and exploits. There are also open companies where you can find, not even on the dark web but on the regular surface web, companies that will pay you over a billion dollars if you find an iOS vulnerability that is zero-click and zero-day. So, this is the situation.

What this book basically says is that we have already seen Stuxnet being used by other countries in Iranian nuclear plants. What stops Iran from using the same on other countries? And they have tried.

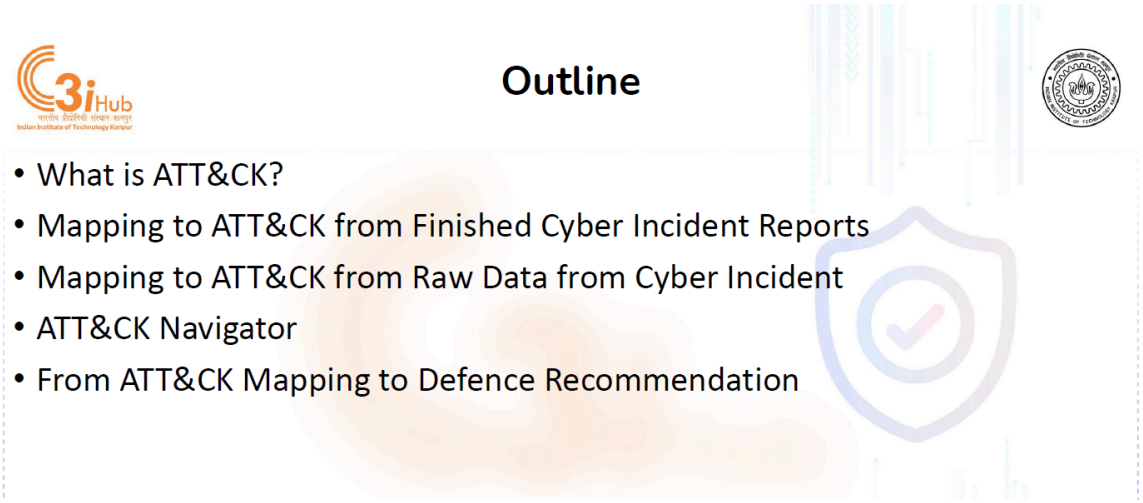
And Iran has actually attacked dams, water systems like the hydro systems in the US. By mistake, they targeted a very small dam, so it didn't cause much damage, but there is a dam with the same name in Oregon. If they had attacked that one, thousands of people could have been flooded away if the gates had opened by remote control. Similarly, the North Koreans are constantly targeting the South Koreans, and the Russians are doing this to Ukraine. They shut down their power and various other systems.

So, what this book is saying is that if we do not have control over this, at some point, we might create a nuclear disaster or cause some kind of weapon system misfiring, leading to

an entire worldwide war and possibly the end of the world. This is a very dystopian view of things. I don't want to scare you, but it is a serious issue to be taken very seriously. I highly suggest reading this book if you can. If you try hard, you will find a PDF copy somewhere on the internet, but I would request you to not use that and instead buy it. It's not very expensive; it's like 500 rupees or something in India.

So, now I'm going to start our new module, MITRE ATT&CK. MITRE ATT&CK is a knowledge base of how adversaries attack our systems. Remember, like in CKC, what we saw is that they presented a very simplistic view, saying there are seven stages through which an adversary has to get into your system, install things, make them permanent and persistent, then communicate with the command and control, and eventually do something harmful, right? MITRE actually came much later than CKC.

They analyzed thousands and thousands of attack incidents, papers, and so on, and they concluded that it is not as simplistic and linear as CKC might suggest. So, they created a knowledge base. This knowledge base is very extensive. It has 14 tactics, over 300 techniques, and even more procedures. This is what is called TTP: tactics, techniques, and procedures.



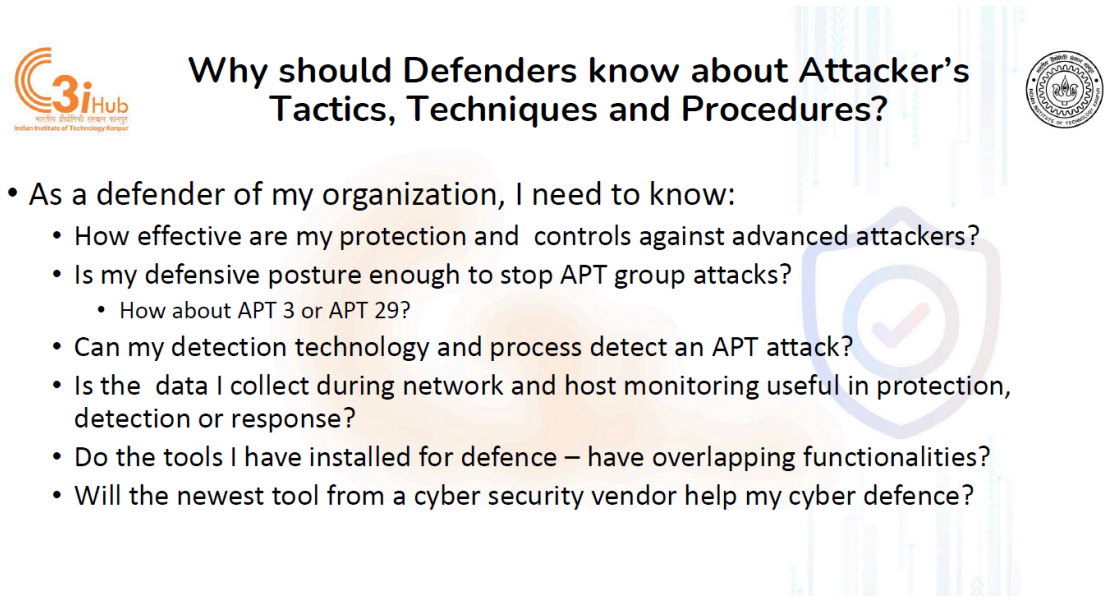
So, we'll talk about what ATT&CK is all about. Then we'll teach you how to map an incident, like an attack incident, into the ATT&CK framework. We can do this from analyst reports or from raw data, the data that we collect as evidence. We'll also discuss a tool that MITRE has provided to help with this kind of work.

This is not to teach you how to attack but for defenders to understand the attacker so that they can figure out for each of these techniques whether their defense is adequate or if they need to do something else.

The goal is to wrap your head around what can happen to your system and figure out how you would stop or detect it when it happens. As a defender, I want to know various things. I want to know whether my current defense is adequate and if the controls I have—like firewalls, endpoint detection, network monitoring, strong authentication, two-factor authentication, network segmentation, and so on—are enough.

The question of 'is this enough' can only be answered if you know what the other side can do. If you assume that the other side is very stupid and will only try phishing and nothing else, then you don't have to do a whole lot. You can stop that by giving a lot of training to your employees and users, telling them not to click on certain things or download suspicious files, and you'll be fine.

But the adversary is not simple. They are much more sophisticated, backed by governments, have a lot of funding, and employ skilled hackers.



So, therefore, I cannot really depend on this small or ad hoc implementation of defense. Sometimes people do not think about adversity and all. They just put a firewall, a proxy, and some antivirus and think that everything is fine, but it is not. You need to actually figure out what the attacker might do and then compare whether you have adequate defense. That is something every defender wants to know.

The second thing is, let us say, I read in the news that educational institutes are now being targeted by APTX. Some APT groups target the health sector, some attack the oil and gas sector, and some target nuclear plants. There could be an APT group attacking the educational sector. As IITK, I would immediately worry about whether I could be a target. If I am the target, I have to read all the information from other incidents—how they got in, what they did, whether they performed any data exfiltration, or whether they carried out a ransomware attack, and so on. Then, I have to check my defense controls to see if I can handle that particular APT.

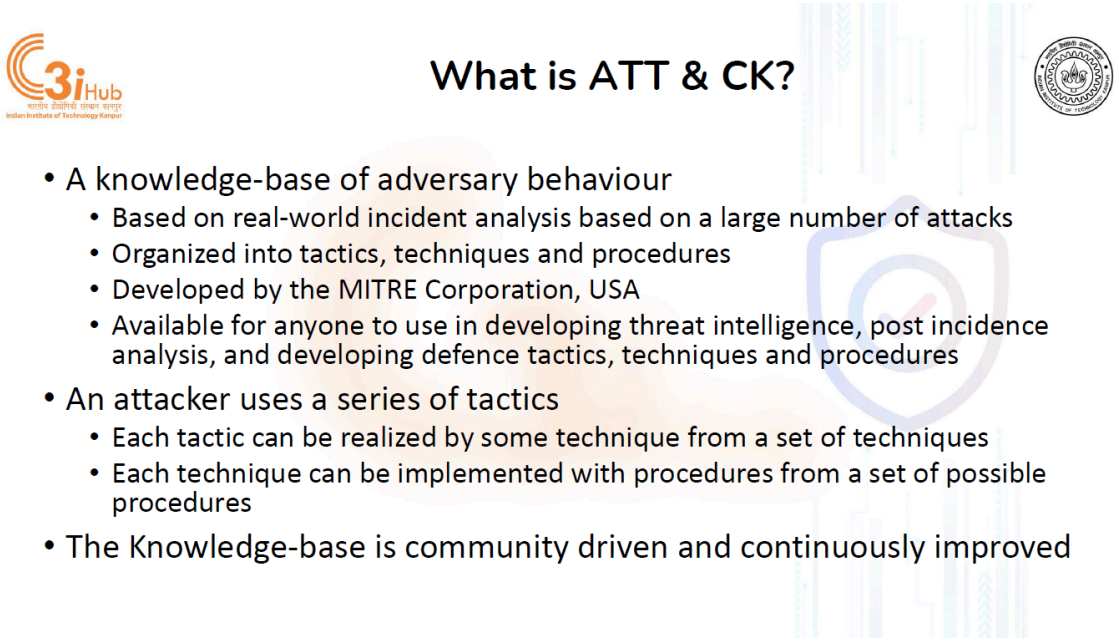
So, this question here is not just about APT3 or APT29—29 is also Russian, and 3 is Chinese. For any kind of threat intelligence that you get in the news or from sources like CERT-IN, indicating that a particular APT group is now focusing on a specific sector, you have to check against your defenses to see if that APT group can be thwarted by your defenses. To do that, you have to understand what that APT group does. The question is, can I stop APT attacks?

Organizations collect a lot of data from their infrastructure, such as network monitoring, endpoint monitoring, logs from all systems, firewall logs, web server logs, and so on. It's a huge amount of data. We analyze it and display the main findings on a screen, like in a

SOC. The question is, is the data I'm collecting useful in protection, detection, or response?

Another question is whether I am actually overdoing it. Do I have many tools with overlapping functionality, meant to detect or defend against the same thing? Maybe I'm unnecessarily buying two different tools and paying the license fees.

When cybersecurity tool vendors come to you, they will tell you all kinds of things. But you have to formulate the right questions in your mind: What is this tool for? With respect to this type of adversarial activity, will this tool help? These questions can be answered better if you formulate everything in terms of MITRE ATT&CK. These are the reasons why MITRE ATT&CK was created. ATT&CK is a knowledge base, not a tool. It's a framework to study the adversary's behavior in a very structured way.



As I said, MITRE Corporation is a think tank. They formed a group that went through a very large number of incidents, analyzing what happened in those incidents and what was done. They came up with a structured way of capturing all these incidents. They said an adversary has a final goal. For example, in the case of Stuxnet, the final goal was to change the program of programmable logic controllers (PLCs) such that the motors rotating the spindles for enriching uranium would sometimes go very fast and sometimes go very slow. Instead of operating at a uniform speed and a critical speed necessary for nuclear enrichment, they had thousands of very large tubes in which uranium was being rotated for enrichment.

These spindles, if they rotate at a critical speed or beyond, only then does it work. That was the whole idea. The attackers figured out that the motors rotate the spindles. Every spindle has a motor, so they decided to target the PLCs, which control the motor speeds.

PLCs are located on the factory floor where the spindles are situated and are controlled by SCADA systems (Supervisory Control and Data Acquisition). PLCs are not like regular computers; they don't have screens or keyboards. Programs for PLCs are created and downloaded from Windows machines. To change the PLC program, attackers needed access to the Windows machines from which the PLC programs were loaded.

These Windows machines were within the network, but the network was not connected to the internet. However, the office network, where regular employees worked, and the network segment containing the Windows machine for PLCs were in the same segment. The attackers got one of the office employees to carry the malware. The malware was carefully written with extensive ground intelligence. They knew exactly which Siemens PLC (S7) was used, how it worked, and how the PLC was loaded with programs from Windows.

They likely used a USB stick with the malware, which an employee brought in and plugged into their machine. The malware copied itself to the machine, and through lateral movement, it eventually reached the machine where the PLC program was loaded. It replaced the PLC program with one that would make the motors run erratically. These sophisticated motors, if run erratically for a while, would burn out. The main idea was to burn out as many motors as possible in a short time, repeatedly causing failures. While motors often crash and burn, the usual failure rate is very low, around 1%.

So, when they started seeing that motors were crashing and burning at a very fast rate, with a very large percentage of motors crashing and burning, it basically halted their uranium enrichment program. They realized something was not right. They analyzed the PLC program and discovered it was a different program, not the original one.

After figuring this out, they realized they had been compromised. At this point, they started noticing Stuxnet everywhere else. Within a couple of months, after the Iranians got to know, Stuxnet was found everywhere in Europe, the US, South America, India, and Asia. Within that year, Stuxnet and its various variants were seen all over the world. The governments that initially launched Stuxnet got really afraid, realizing they had unleashed something that couldn't be contained.

They thought they would just use it to delay Iran's nuclear program without anyone knowing how so many motors were malfunctioning. By the time the issue was discovered, there would be a significant delay in the advancement of Iran's nuclear

program. Unfortunately, the malware was exposed, leading to the creation of various Stuxnet variants and other malware from the same group that initiated Stuxnet.

I have a whole lecture on that, which I will post. The idea I'm trying to convey is that the adversary has an eventual goal, which in this case was to destroy Iran's nuclear capabilities by delaying it. That is their final goal. But they don't achieve the final goal directly. You cannot attain the final goal directly. You have to set various short-range goals.

How do I get in? The system is not connected to the internet, and they don't read emails on their computer, so I cannot phish them.

So, I have to figure out how to deliver the malware. USB it is, right? Once they got that, they achieved one goal. But before doing that, they also had to do weaponization, because writing the Stuxnet worm was a lot of work, probably years of work. So, weaponization was done. And then reconnaissance was done to identify which executives to target. This step is essential.

Reconnaissance was done, and weaponization was done. In fact, reconnaissance was probably done after weaponization. You write Stuxnet in the lab, test it on a test bed, and then figure out who in that particular facility would be amenable to taking a USB inside without suspecting anything. That is reconnaissance. Then comes the delivery. Delivery was through the USB stick. The worm has to figure out the machine in which it was initially executed.

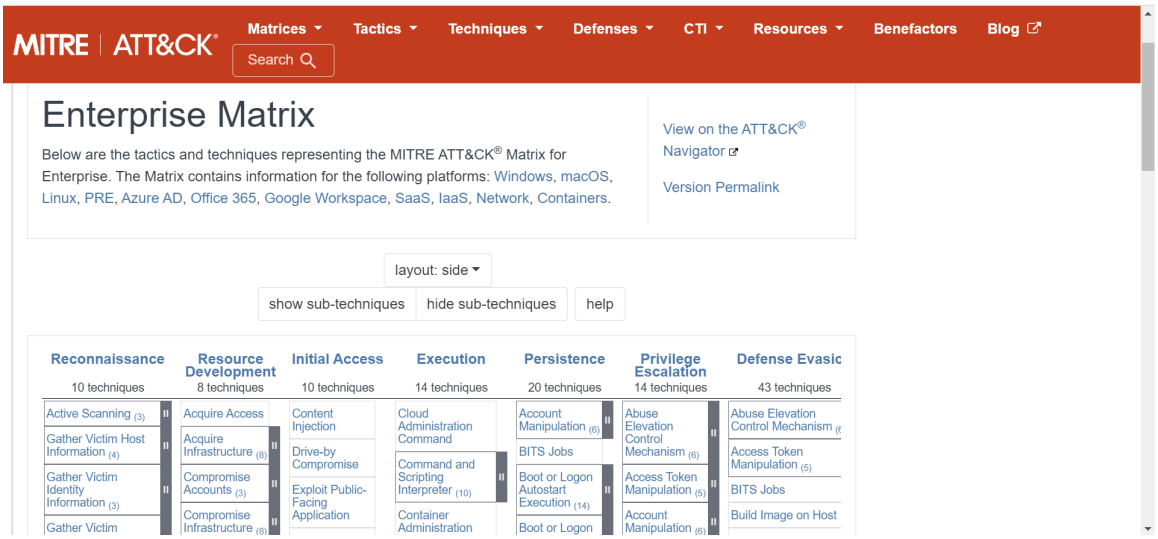
It may not be a high-privilege account, so it has to figure out how to perform privilege escalation or move across that machine to another machine, eventually finding the one that has the PLC system. These are small goals: how to get in, how to move from one machine to another, and how to collect data about which machine has the right target system. All these tasks are tactics. When an attacker wants to achieve a goal, they string together tactics.

Tactics do not necessarily occur in a linear order; they may happen in multiple orders. The same tactic may be used multiple times in the same chain of events, but eventually, the final goal is achieved. To implement the tactics, you need techniques. There are many different techniques. For example, delivery by USB is one technique for delivery, but you could also deliver it through a CD, email, or by finding a weakness in their local network and sending a spy into the facility.

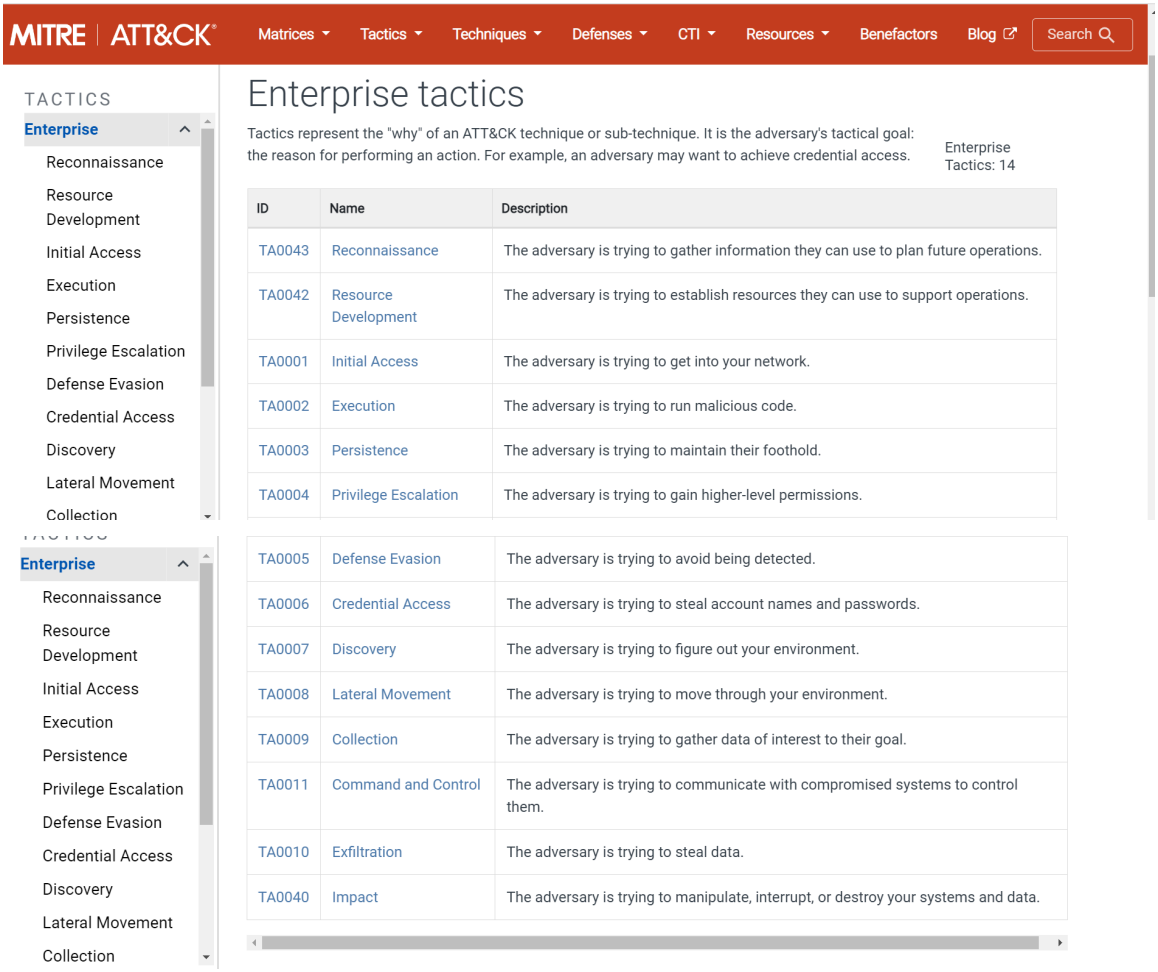
Each tactic has multiple techniques by which it can be realized, and techniques can be described in terms of procedures—how a technique is actually applied. This knowledge

base is community-driven; it's not solely done by MITRE people. They invited everyone to contribute, and it has become a huge and very useful knowledge base for everyone.

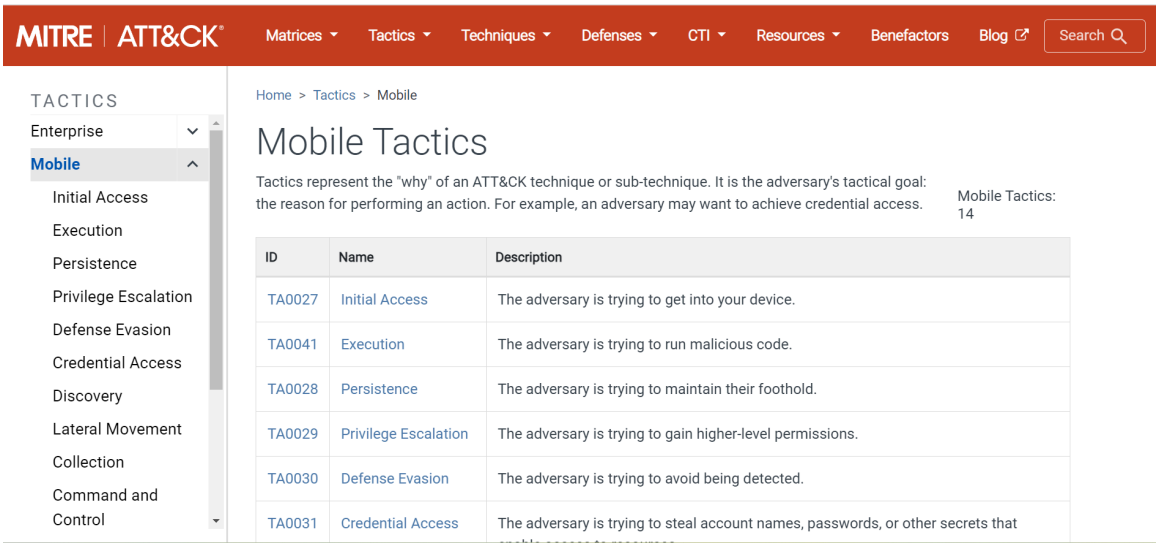
Okay, let me show you the knowledge base because I don't think I have a whole lot of time here, just five minutes.

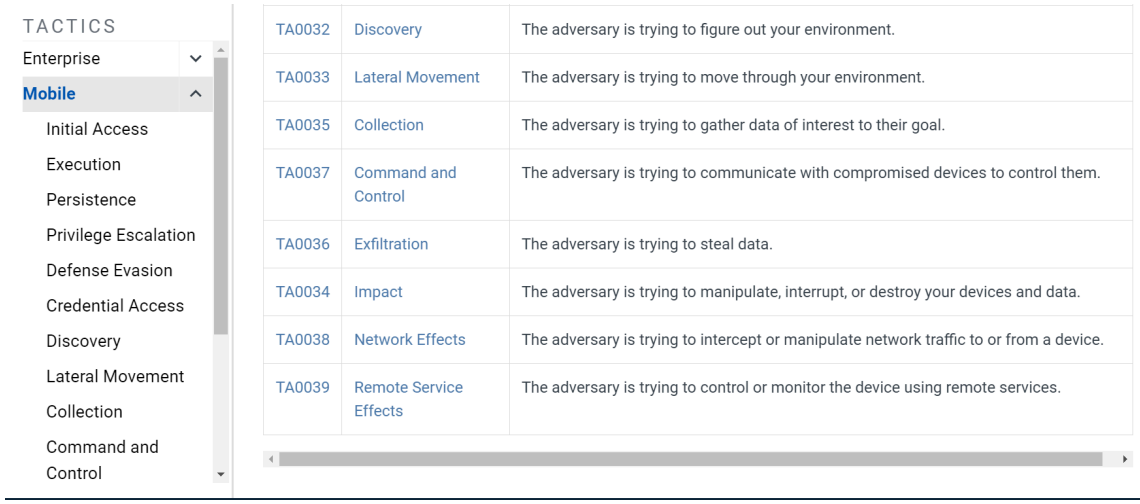


So, attack.mitre.org is where you have to go. Here, you will see the tactics. The tactics that I'm going to talk about in the class are enterprise tactics. There are 14 enterprise tactics.

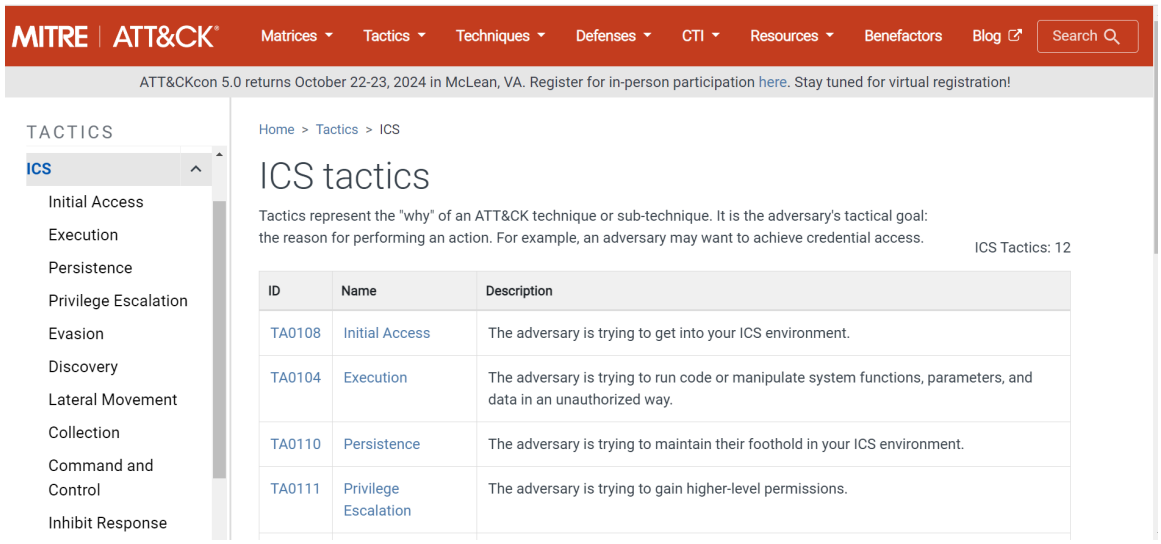


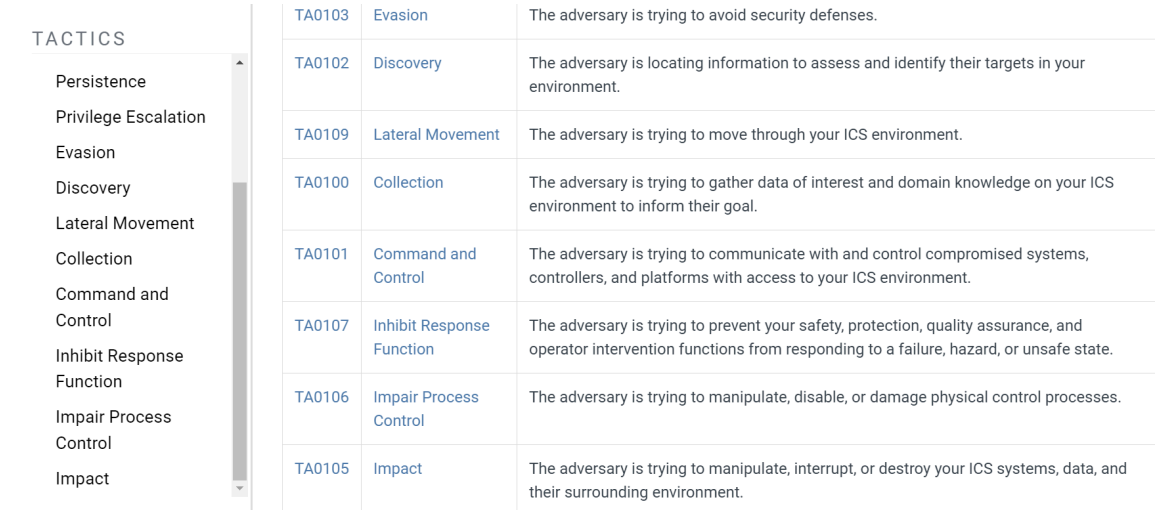
Now, if you go into their mobile tactics, you will see a slightly different set of tactics for how attacks on mobile phones happen.





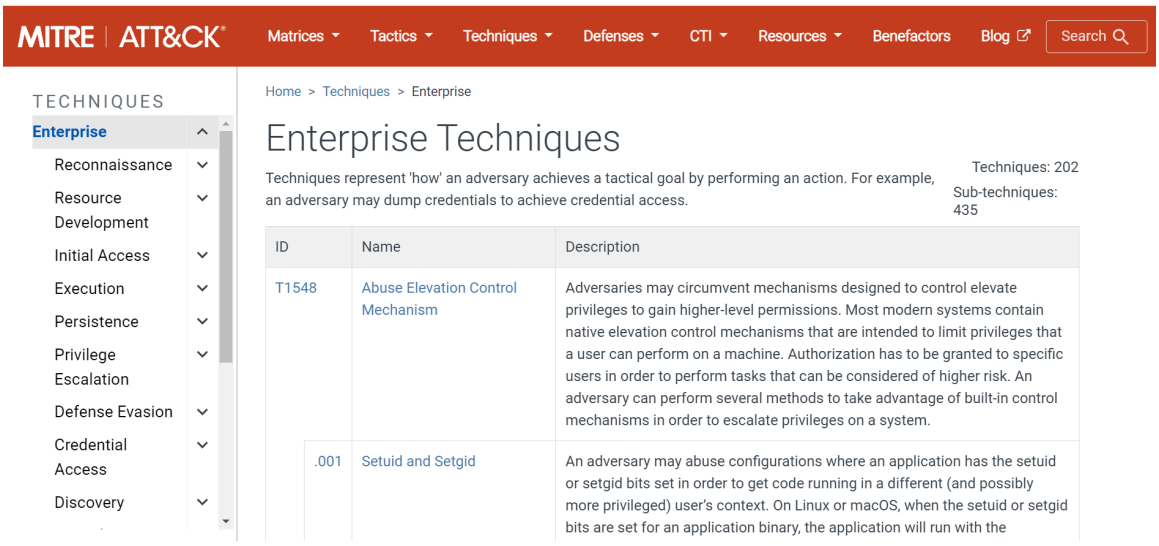
And if you go to their ICS tactics—that's the industrial control system—you will see a slightly different and smaller number of tactics. This doesn't mean that attacks on ICS require fewer tactics, but rather that the attacks analyzed so far have revealed only these tactics.

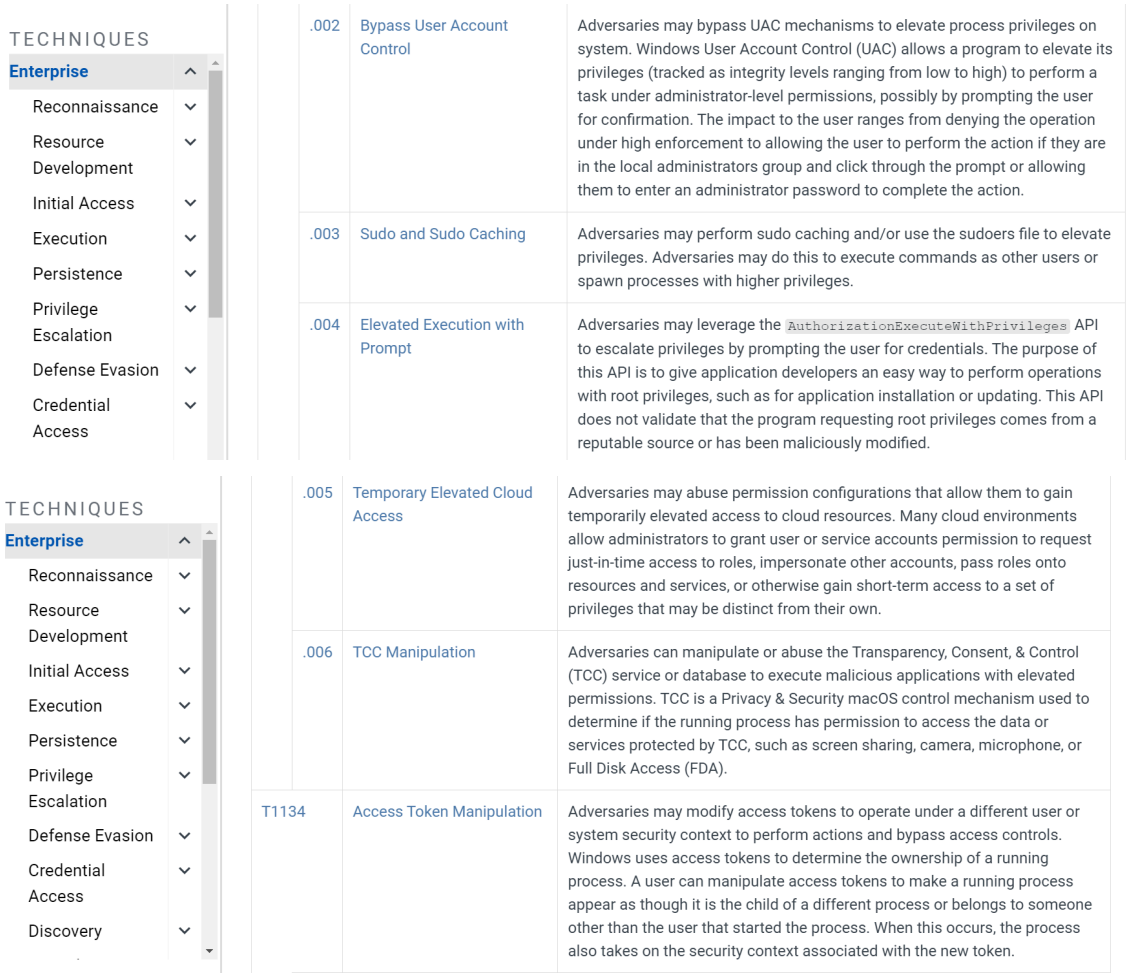




Tomorrow, there may be another tactic added to this list, but so far these are the tactics that have been seen in use.

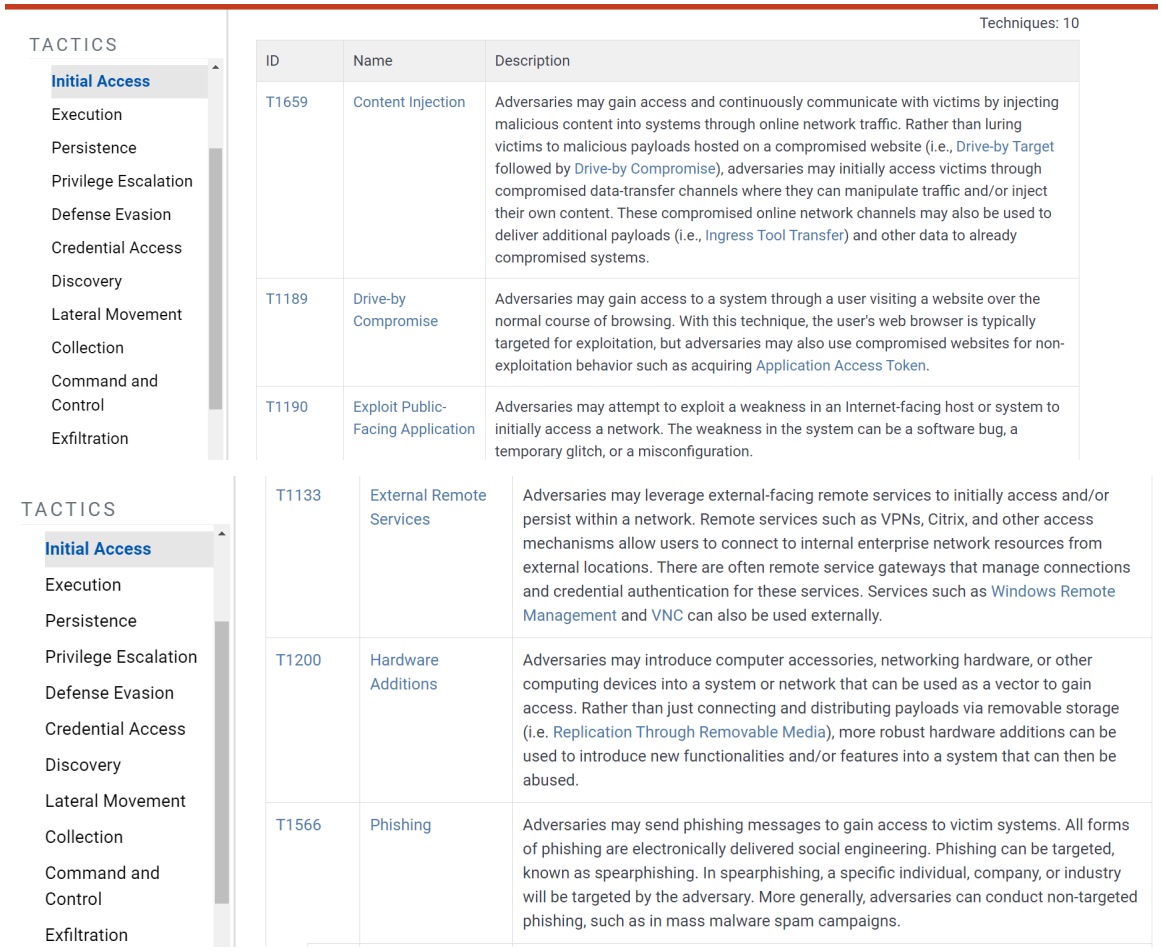
Then, if you go into the techniques, such as enterprise techniques, you will see a list of techniques. There are 300 plus techniques.



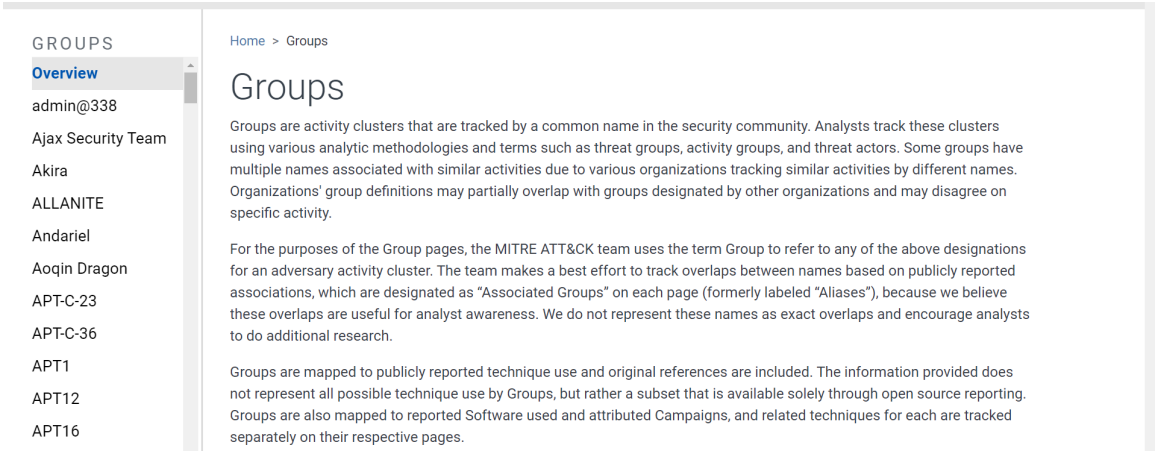


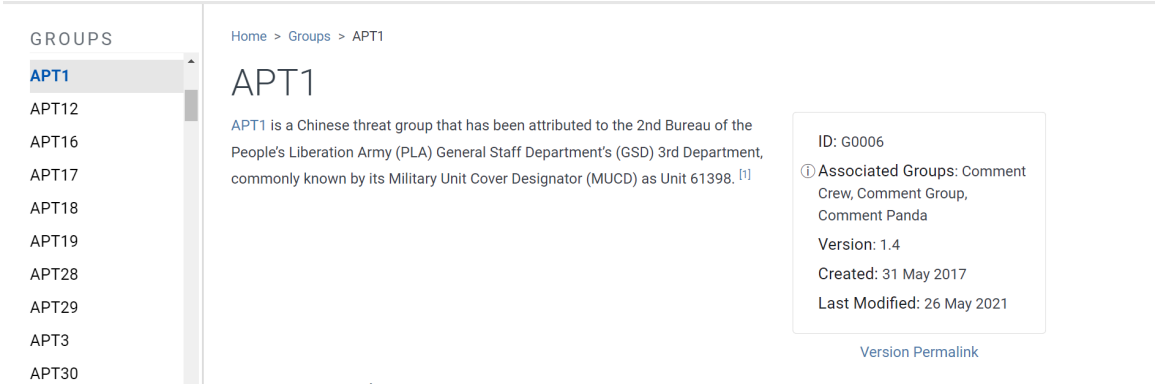
So here, and then there are sub-techniques. For example, under 'abuse elevation control mechanism,' which is about privilege escalation, there are multiple different sub-techniques. Those of you who know about set UID and set GID, here is the bypass of user account control, using sudo or sudo caching, elevated execution with prompt, temporary elevated cloud access, access token manipulation. There are many different ways you can actually perform privilege escalation.

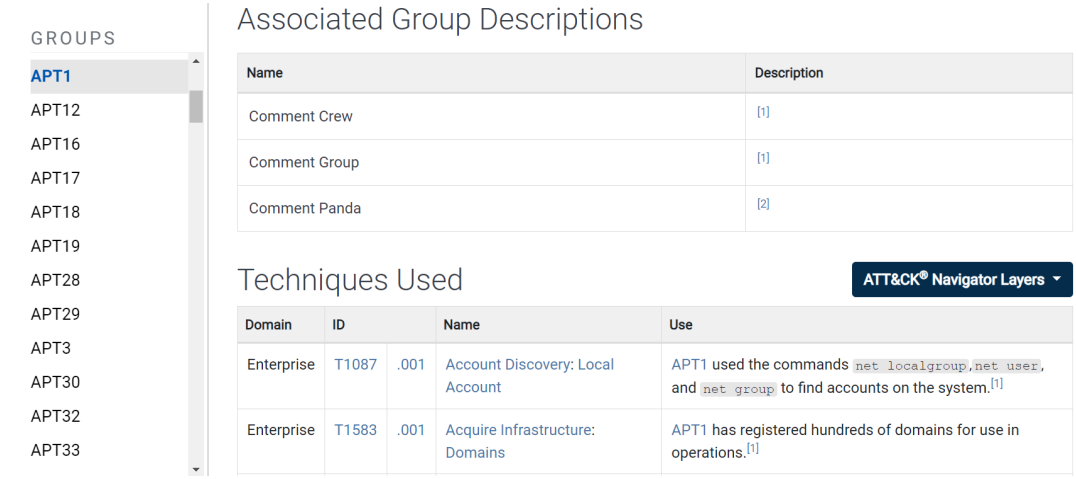
Similarly, you can have techniques associated with, let's say, initial access. For initial access, you can have content injection, drive-by compromise, exploit public-facing application, external remote services, and so on.



So these are techniques. We'll get into this later, but let's talk about threat intelligence. There are threat groups, and you can see all these different threat groups. For example, APT-1 is a Chinese threat group attributed to the 2nd Bureau of the People's Liberation Army General Staff Department's 3rd Department, commonly known by its military unit cover designator as Unit 61398. This group has been analyzed quite a bit by various threat intelligence agencies.







So, that is why they are being so specific about who might be behind APT-1. Some of the groups may not be known that definitively. Here you have all the techniques that have been seen to be used by this APT group and the kind of software they use for their attacks. This is where you find more information about APT groups. You can also find information about different software used for attacks

GROUPS

APT1
APT12
APT16
APT17
APT18
APT19
APT28
APT29
APT3
APT30
APT32
APT33

Software

ID	Name	References	Techniques
S0017	BISCUIT	[1]	Command and Scripting Interpreter: Windows Command Shell, Encrypted Channel: Asymmetric Cryptography, Fallback Channels, Ingress Tool Transfer, Input Capture: Keylogging, Process Discovery, Screen Capture, System Information Discovery, System Owner/User Discovery, System Time Discovery
S0119	Cachedump	[1]	OS Credential Dumping: Cached Domain Credentials
S0025	CALENDAR	[1]	Command and Scripting Interpreter: Windows Command Shell, Web Service: Bidirectional Communication
S0026	GLOOXMAIL	[1]	Web Service: Bidirectional Communication
S0008	gsecdump	[1]	OS Credential Dumping: Security Account Manager, OS Credential Dumping: LSA Secrets
S0100	ipconfig	[1]	System Network Configuration Discovery

GROUPS

APT1
APT12
APT16
APT17
APT18
APT19
APT28
APT29
APT3
APT30
APT32
APT33

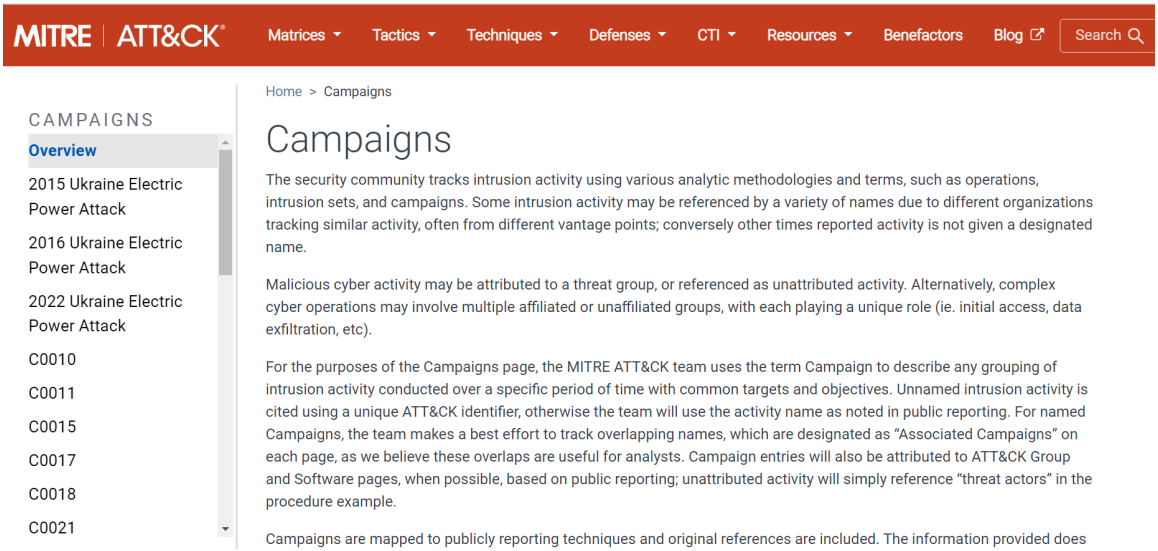
S0121	Lslsass	[1]	OS Credential Dumping: LSASS Memory
S0002	Mimikatz	[1]	Access Token Manipulation: SID-History Injection, Account Manipulation, Boot or Logon Autostart Execution: Security Support Provider, Credentials from Password Stores, Credentials from Password Stores: Credentials from Web Browsers, Credentials from Password Stores: Windows Credential Manager, OS Credential Dumping: DCSync, OS Credential Dumping: Security Account Manager, OS Credential Dumping: LSASS Memory, OS Credential Dumping: LSA Secrets, Rogue Domain Controller, Steal or Forge Authentication Certificates, Steal or Forge Kerberos Tickets: Golden Ticket, Steal or Forge Kerberos Tickets: Silver Ticket, Unsecured Credentials: Private Keys, Use Alternate Authentication Material: Pass the Hash, Use Alternate Authentication Material: Pass the Ticket
S0039	Net	[1]	Account Discovery: Domain Account, Account Discovery: Local Account, Create Account: Local Account, Create Account: Domain Account, Indicator Removal: Network Share Connection Removal, Network Share Discovery, Password Policy Discovery, Permission Groups Discovery: Domain Groups, Permission Groups Discovery: Local Groups, Remote Services: SMB/Windows Admin Shares, Remote System Discovery, System Network Connections Discovery, System Service Discovery, System Services: Service Execution, System Time Discovery

GROUPS

APT1
APT12
APT16
APT17
APT18
APT19
APT28
APT29
APT3
APT30
APT32
APT33

S0122	Pass-The-Hash Toolkit	[1]	Use Alternate Authentication Material: Pass the Hash
S0012	PoisonIvy	[1]	Application Window Discovery, Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder, Boot or Logon Autostart Execution: Active Setup, Command and Scripting Interpreter: Windows Command Shell, Create or Modify System Process: Windows Service, Data from Local System, Data Staged: Local Data Staging, Encrypted Channel: Symmetric Cryptography, Ingress Tool Transfer, Input Capture: Keylogging, Modify Registry, Obfuscated Files or Information, Process Injection: Dynamic-link Library Injection, Rootkit
S0029	PsExec	[1]	Create Account: Domain Account, Create or Modify System Process: Windows Service, Lateral Tool Transfer, Remote Services: SMB/Windows Admin Shares, System Services: Service Execution
S0006	pwdump	[1]	OS Credential Dumping: Security Account Manager
S0345	Seasalt	[3][5]	Application Layer Protocol: Web Protocols, Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder, Command and Scripting Interpreter: Windows Command Shell, Create or Modify System Process: Windows Service, File and Directory Discovery, Indicator Removal: File Deletion, Ingress Tool Transfer,

So you can see, and this list continues to grow as we learn more. There are also campaigns, which are basically the operations that APT groups carry out. For example, if you want to know about the 2015 electric power attack, you can go here and see what techniques were used.



And from that, you can figure out what tactics were used. You can see some of the software that was used for the Ukraine electric power grid attack and the various techniques that were employed.

CAMPAIGNS

- 2015 Ukraine Electric Power Attack
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

https://attack.mitre.org/campaigns/C0028

Home > Campaigns > 2015 Ukraine Electric Power Attack

2015 Ukraine Electric Power Attack

2015 Ukraine Electric Power Attack was a Sandworm Team campaign during which they used BlackEnergy (specifically BlackEnergy3) and KillDisk to target and disrupt transmission and distribution substations within the Ukrainian power grid. This campaign was the first major public attack conducted against the Ukrainian power grid by Sandworm Team.

ID: C0028
First Seen: December 2015 [1]
Last Seen: January 2016 [1]
Version: 1.0
Created: 27 September 2023
Last Modified: 06 October 2023

Version Permalink

Groups

ID	Name	Description
G0034	Sandworm Team	[2] [3]

CAMPAIGNS

- 2015 Ukraine Electric Power Attack
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

Groups

ID	Name	Description
G0034	Sandworm Team	[2] [3]

Techniques Used

ATT&CK® Navigator Layers

Domain	ID		Name	Use
Enterprise	T1071	.001	Application Layer Protocol: Web Protocols	During the 2015 Ukraine Electric Power Attack, Sandworm Team used BlackEnergy to communicate between compromised hosts and their command-and-control servers via HTTP post requests. [1]
Enterprise	T1059	.005	Command and Scripting Interpreter: Visual Basic	During the 2015 Ukraine Electric Power Attack, Sandworm Team installed a VBA script called `vba_macro.exe`. This macro dropped `FONTCACHE.DAT`, the primary BlackEnergy implant; `rundll32.exe`, for executing the malware; `NTUSER.log`, an empty file; and desktop.ini, the default file used to determine folder displays on Windows machines. [1]

CAMPAIGNS

- 2015 Ukraine Electric Power Attack
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

Enterprise	T1136	.002	Create Account: Domain Account	During the 2015 Ukraine Electric Power Attack, Sandworm Team created privileged domain accounts to be used for further exploitation and lateral movement. [1]
Enterprise	T1133		External Remote Services	During the 2015 Ukraine Electric Power Attack, Sandworm Team installed a modified Dropbear SSH client as the backdoor to target systems. [1]
Enterprise	T1562	.001	Impair Defenses: Disable or Modify Tools	During the 2015 Ukraine Electric Power Attack, Sandworm Team modified in-registry internet settings to lower internet security. [1]
Enterprise	T1070	.004	Indicator Removal: File Deletion	During the 2015 Ukraine Electric Power Attack, vba_macro.exe deletes itself after `FONTCACHE.DAT`, `rundll32.exe`, and the associated .lnk file is delivered. [1]
Enterprise	T1105		Ingress Tool Transfer	During the 2015 Ukraine Electric Power Attack, Sandworm Team pushed additional malicious tools onto an infected system to steal user credentials, move laterally, and destroy data. [1]
Enterprise	T1056	.001	Input Capture:	During the 2015 Ukraine Electric Power Attack, Sandworm Team

CAMPAIGNS

- **2015 Ukraine Electric Power Attack**
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

Enterprise	T1056	.001	Input Capture: Keylogging	During the 2015 Ukraine Electric Power Attack, Sandworm Team gathered account credentials via a BlackEnergy keylogger plugin. [1][4]
Enterprise	T1570		Lateral Tool Transfer	During the 2015 Ukraine Electric Power Attack, Sandworm Team moved their tools laterally within the corporate network and between the ICS and corporate network. [1]
Enterprise	T1112		Modify Registry	During the 2015 Ukraine Electric Power Attack, Sandworm Team modified in-registry Internet settings to lower internet security before launching `rundll32.exe`, which in-turn launches the malware and communicates with C2 servers over the Internet. [1].
Enterprise	T1040		Network Sniffing	During the 2015 Ukraine Electric Power Attack, Sandworm Team used BlackEnergy's network sniffer module to discover user credentials being sent over the network between the local LAN and the power grid's industrial control systems. [5]
Enterprise	T1566	.001	Phishing: Spearphishing Attachment	During the 2015 Ukraine Electric Power Attack, Sandworm Team obtained their initial foothold into many IT systems using Microsoft Office attachments delivered through phishing emails. [4]

CAMPAIGNS

- **2015 Ukraine Electric Power Attack**
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

Enterprise	T1055		Process Injection	During the 2015 Ukraine Electric Power Attack, Sandworm Team loaded BlackEnergy into svchost.exe, which then launched iexplore.exe for their C2. [1]
Enterprise	T1018		Remote System Discovery	During the 2015 Ukraine Electric Power Attack, Sandworm Team remotely discovered systems over LAN connections. OT systems were visible from the IT network as well, giving adversaries the ability to discover operational assets. [5]
Enterprise	T1218	.011	System Binary Proxy Execution: Rundll32	During the 2015 Ukraine Electric Power Attack, Sandworm Team used a backdoor which could execute a supplied DLL using `rundll32.exe`. [1]
Enterprise	T1204	.002	User Execution: Malicious File	During the 2015 Ukraine Electric Power Attack, Sandworm Team leveraged Microsoft Office attachments which contained malicious macros that were automatically executed once the user permitted them. [4]
Enterprise	T1078		Valid Accounts	During the 2015 Ukraine Electric Power Attack, Sandworm Team used valid accounts on the corporate network to escalate privileges, move laterally, and establish persistence within the corporate network. [4]

CAMPAIGNS

- **2015 Ukraine Electric Power Attack**
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

ICS	T0803	Block Command Message	During the 2015 Ukraine Electric Power Attack, Sandworm Team blocked command messages by using malicious firmware to render serial-to-ethernet converters inoperable. [4]
ICS	T0804	Block Reporting Message	During the 2015 Ukraine Electric Power Attack, Sandworm Team blocked reporting messages by using malicious firmware to render serial-to-ethernet converters inoperable. [4]
ICS	T0805	Block Serial COM	During the 2015 Ukraine Electric Power Attack, Sandworm Team overwrote the serial-to-ethernet converter firmware, rendering the devices not operational. This meant that communication to the downstream serial devices was either not possible or more difficult. [1]
ICS	T0885	Commonly Used Port	During the 2015 Ukraine Electric Power Attack, Sandworm Team used port 443 to communicate with their C2 servers. [1]
ICS	T0884	Connection Proxy	During the 2015 Ukraine Electric Power Attack, Sandworm Team established an internal proxy prior to the installation of backdoors within the network. [1]

CAMPAIGNS

- **2015 Ukraine Electric Power Attack**
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

ICS	T0813	Denial of Control	During the 2015 Ukraine Electric Power Attack, KillDisk rendered devices that were necessary for remote recovery unusable, including at least one RTU. Additionally, Sandworm Team overwrote the firmware for serial-to-ethernet converters, denying operators control of the downstream devices. [1][4]
ICS	T0814	Denial of Service	During the 2015 Ukraine Electric Power Attack, power company phone line operators were hit with a denial of service attack so that they couldn't field customers' calls about outages. Operators were also denied service to their downstream devices when their serial-to-ethernet converters had their firmware overwritten, which bricked the devices. [4]
ICS	T0816	Device Restart/Shutdown	During the 2015 Ukraine Electric Power Attack, Sandworm Team scheduled the uninterruptable power supplies (UPS) to shutdown data and telephone servers via the UPS management interface. [4] [1]
ICS	T0822	External Remote Services	During the 2015 Ukraine Electric Power Attack, Sandworm Team used Valid Accounts taken from the Windows Domain Controller to access the control system Virtual Private Network (VPN) used by grid operators. [1]

CAMPAIGNS

- **2015 Ukraine Electric Power Attack**
- 2016 Ukraine Electric Power Attack
- 2022 Ukraine Electric Power Attack
- C0010
- C0011
- C0015
- C0017
- C0018
- C0021
- C0026

Software

ID	Name	Description
S0089	BlackEnergy	[1]
S0607	KillDisk	[1]

References

1. Booz Allen Hamilton When The Lights Went Out Retrieved. 2019/10/22
2. Andy Greenberg. (2017, June 28). How an Entire Nation Became Russia's Test Lab for Cyberwar. Retrieved September 27, 2023.
3. Scott W. Brady. (2020, October 15). United States vs. Yuriy Sergeyevich Andrienko et al.. Retrieved November 25, 2020.
4. Electricity Information Sharing and Analysis Center; SANS Industrial Control Systems. (2016, March 18). Analysis of the Cyber Attack on the Ukranian Power Grid: Defense Use Case. Retrieved March 27, 2018.
5. Charles McLellan. (2016, March 4). How hackers attacked Ukraine's power grid: Implications for Industrial IoT security. Retrieved September 27, 2023.