2024 DUBLIN

2 - 4 October, 2024 / Dublin, Ireland

LIFE AND DEATH: BUILDING DETECTION, FORENSICS, AND INTELLIGENCE AT SCALE

Selena Larson & Konstantin Klinger

Proofpoint, USA

slarson@proofpoint.com
kklinger@proofpoint.com

ABSTRACT

How can threat intelligence and detection teams build a collaborative, productive relationship and improve overall organizational security? Our paper will focus on how detection engineering and threat hunting (DEaTH) informs detection pipelines to enable automated forensic analysis of the biggest threats to enterprises, and how security practitioners can use this type of data to improve their own defence.

INTRODUCTION

Detection engineering and threat hunting (DEaTH) is not just a delightful acronym. It represents a crucial function in security teams. Often, the teams responsible for detections, threat hunting, and intelligence analysis are siloed within an organization. But increasingly, these functions are coming together to develop holistic, intelligence-driven tooling and capabilities for defence.

Detection engineering and threat hunting (DEaTH), together with cyber threat intelligence (CTI), informs detection pipelines to enable automated forensic analysis of the biggest threats to enterprises, and security practitioners can use this type of data to improve defence. By enabling threat research, intelligence, and detection teams with the appropriate tools and resources, organizations can build a collaborative, productive relationship and improve overall organizational security.

Within our organization each day, *Proofpoint* analyses billions of emails, and millions of those are identified as malicious. In this paper we will discuss how to work together with cross-functional teams to identify new techniques and create detections to prevent threat actor exploitation at scale, while automating configuration and forensic extraction, and MITRE ATT&CK mapping for end-users.

THE DETECTION PIPELINE

No organization or pipeline is the same, so this paper will focus on what we do at *Proofpoint*. Hopefully this will provide a blueprint by which others can identify similarities in their own workflows. The concept we follow is DDX, which stands for detection, detonation, and configuration extraction. From the beginning of an attack chain all the way through malware detonation, we try to detect and block maliciousness at every level.

For example, when an email is observed, there are various characteristics that can be identified as malicious, such as the headers, body text, content sentiment, embedded graphics, as well as URLs or attachments. These are then sent through our static and dynamic engines to generate results of automated extraction and forensic analysis including identifying the malware family, configuration and behaviours.

Static detections involve the evaluation and examination of a given file, URL, or artefact without executing it. Dynamic detections focus on the behaviour of the analysis target during its execution. After the email message itself is examined, extracted URLs and files go into our static and dynamic detection engines. Time is a critical resource as email is considered real-time traffic. Therefore, an email cannot be held for an unlimited amount whilst deciding whether to block it or not. This means the engines must be quick to return a verdict, and we cannot wait for the dynamic engine results for every URL or file.

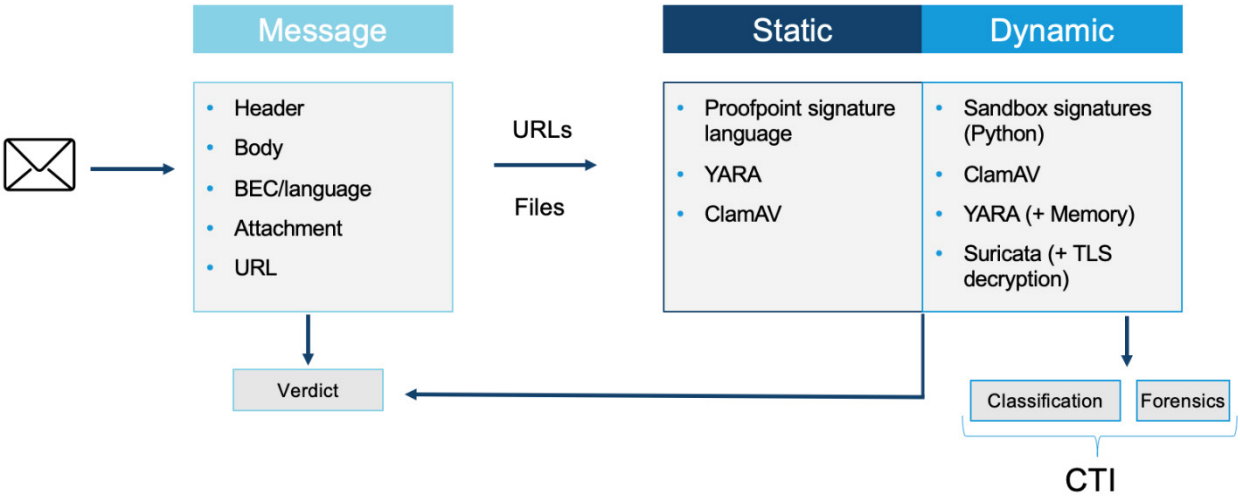

Figure 1: Proofpoint detection pipeline.

Our detection process follows the pyramid of pain concept [1] and the defence in depth concept [2], which means we are trying to detect things at the message level, static level, and dynamic sandbox level. Further, we are not only trying to create detection on atomic indicators of compromise (IOCs) such as hashes and URLs, but also trying to detect malicious

content based on tactics, techniques and procedures (TTPs) used and the actual behaviour of malware independent of static attributes.

It could stop there, because the threats are blocked, and people are protected. However, one of the most crucial parts of detection is the resulting information that is surfaced to build out a full picture of a malicious campaign.

A campaign is defined as a time-bound set of related threat activity. This implies, even in cases where no attribution is made, that the threats from a given campaign result from attacks perpetrated by the same threat actor. Threat actors are tracked groups that are defined by their malware, behaviour, targeting, and overall activity. *Proofpoint* uses a numbering system to define them, for example TA577. Threats identified as part of a campaign may be related by a variety of factors including distribution or hosting infrastructure, overlap in message forensics such as header components, a common payload, or other facets.

Forensics extracted from detection functions are crucial for threat researchers and intelligence analysts to build a full picture of threat activity. They are used for things like attribution, threat actor tracking, pivoting, additional threat hunting, and other facets of CTI functions. Further, incident responders in targeted organizations with successful breaches can use the extracted forensics for incident response and the blocking / sinkholing of indicators of compromise (IOCs) as an additional protection layer.

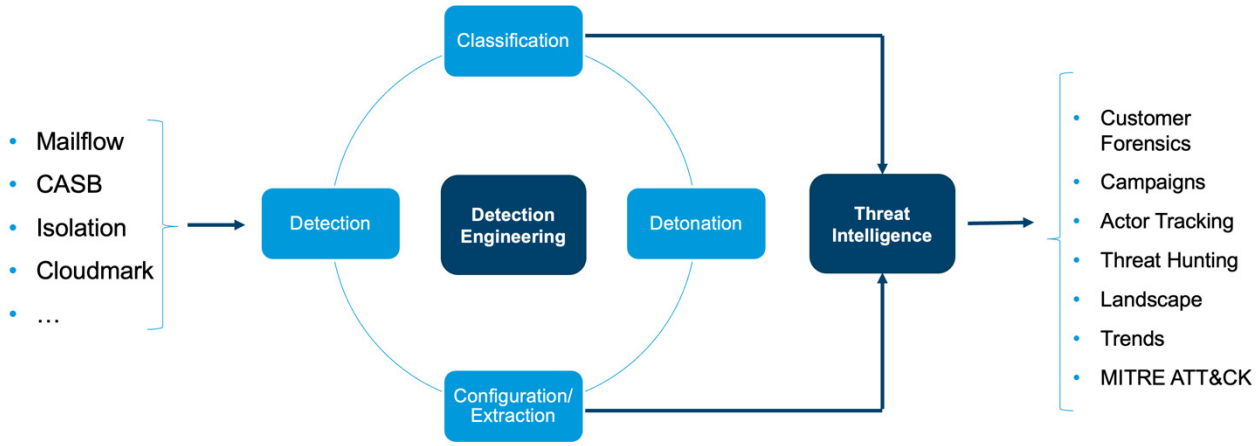

Figure 2: Cyber threat intelligence (CTI) and DEaTH cycle.

EXAMPLES

The necessity of a close relationship between intelligence and detection functions within an organization is due to the fact that the threat landscape is increasingly dynamic, requiring defenders to be proactive in identifying, monitoring and responding to threats. In the same way a cyber intelligence function relies on detection teams, DEaTH should be informed and driven by cyber intelligence, especially in the cybercrime threat landscape where threat actors are constantly developing [3] new TTPs to evade defences, which then forces defenders to respond quickly.

TA577 (cybercrime)

One example illustrating the close collaboration between CTI and detection engineering efforts can be found in the work against TA577. TA577 is one of the most sophisticated and persistent cybercriminal threats and has been tracked by our researchers since 2020. It is an initial access broker and was a main Qbot affiliate before the malware's disruption in summer 2023 [4]. Since then, the actor has experimented with numerous new techniques, attack chains, file types and malware. In some cases, the actor A/B tests attack chains, with a morning wave of malspam leading to one attack chain, while the afternoon wave uses completely different TTPs.

Threat researchers continuously monitor TA577's activity and work closely with the detection team to identify gaps as soon as new activity is identified. Researchers track TA577 campaigns via various attributes including lures, filenames and URL patterns, various email metadata and characteristics, attack chains, malware, and other data. Since 2023, TA577 has demonstrated varied and dynamic attack chains with as many as 10 unique attack chains per month.

In February 2024, *Proofpoint* identified a TA577 campaign using two different Java droppers to deliver Pikabot. In this campaign, emails contained a password-protected ZIP file with a .jar Java archive file inside, as well as a random data file which did not have a purpose in the attack chain and was likely used to increase the size of the ZIP container in an attempt to bypass detection mechanisms.

The two Java dropper variants included:

- Variant 1, which had the Pikabot DLL embedded in the Java file itself.

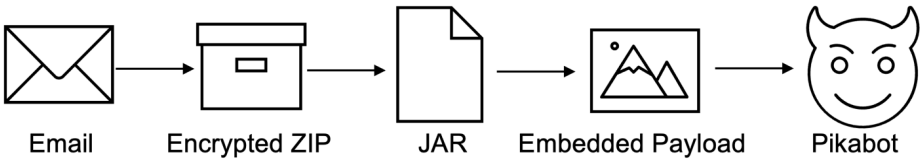

Figure3: Graphic demonstrating Variant 1 attack chain.

- Variant 2, which downloaded the Pikabot payload from a URL.

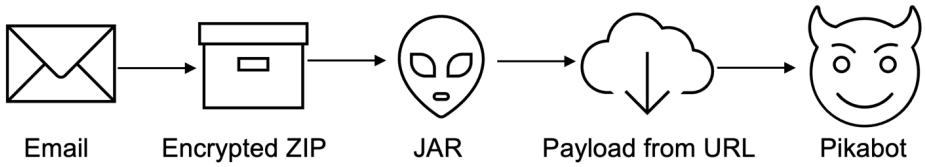

Figure 4: Graphic demonstrating Variant 2 attack chain.

Notably, TA577 used both variants within the same campaign, which researchers suspect was an example of A/B testing.

Focusing on the second variant, the following walk-through demonstrates the previously mentioned concepts of DDX, pyramid of pain, and defence in depth based on this example.

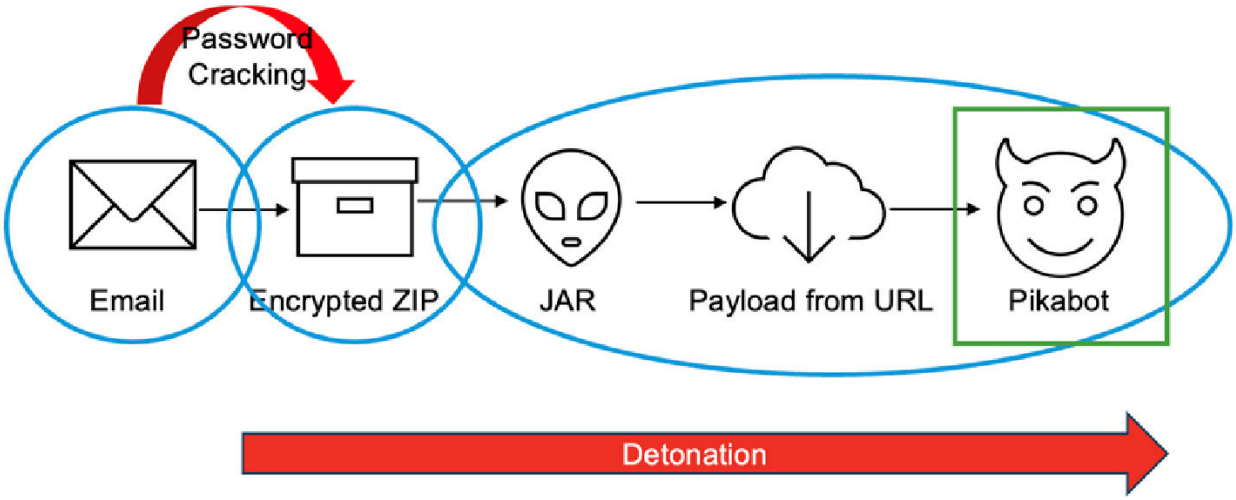

Figure 5: DDX (Detection, Detonation and Configuration Extraction) flow for variant 2.

Message

At the message level, it's already possible to identify an email as TA577 due to the attacker's typical reuse of specific file and URL naming patterns, certain email characteristics, and their love for the thread hijacking [5] technique. Email artefacts such as headers alone are often not enough to determine maliciousness – with thread hijacking, for example, the email can look like it's coming from a legitimate conversation or account. Threat actors can also use compromised mail infrastructure for delivery, which masquerades as legitimate traffic. But when combined with things like email body content, URLs or attachments, and volume/targeting, it can be used for identification of threats.

ZIP

Further, a Java executable within an encrypted ZIP container is unusual for most people to receive via email. It is possible to define so-called policy kills to already disallow that by policy settings.

AlienLoader

Let's take a look at the actual Java dropper, which *Proofpoint* researchers dubbed AlienLoader due to the popular threat label name from *VirusTotal* [6]. Looking at the decompiled binary and its behaviour in a sandbox run identifies numerous detection opportunities.

The sample reaches out to the payload URL and tries to create and write the downloaded payload to a file under 'C:\Users\Public\filename.exe' before it sleeps for 15,000 milliseconds. After that, it starts 'filename.exe' in a new process. Chaining this

behaviour together in a state machine results in a dynamic detection signature for this loader. But a static YARA rule can also be created as an additional layer if detonation fails. A sandbox could detect the loader statically and dynamically based on observed behaviours, but in addition to that, researchers also observe the Pikabot payload downloaded via the URL in the dynamic sandbox run. That behaviour can be detected with network signatures as well as with generic sandbox signatures.

```
import java.io.BufferedInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.net.URL;
import java.net.URLConnection;

public class uPXz5t8M {
    public static void main(String[] paramArrayOfString) {
        try {
            String str1 = "http://94.156.8.9/PE1BD/186304";
            String str2 = "C:\\users\\public\\filename.exe";
            try {
                downloadFile(str1, str2);
            } catch (IOException iOException) {}
            Thread.sleep(15000L);
            Runtime.getRuntime().exec("c:\\users\\public\\filename.exe");
        } catch (Exception exception) {
            System.out.println("Error");
        }
    }

    private static void downloadFile(String paramString1, String paramString2) throws IOException {
        URL uRL = new URL(paramString1);
        URLConnection uRLConnection = uRL.openConnection();
        try(BufferedInputStream null = new BufferedInputStream(uRLConnection.getInputStream());
                FileOutputStream null = new FileOutputStream(paramString2)) {
            byte[] arrayOfByte = new byte[1024];
            int i;
            while ((i = bufferedInputStream.read(arrayOfByte, 0, 1024)) != -1)
                fileOutputStream.write(arrayOfByte, 0, i);
        }
    }
}
```

Figure 6: Decompiled AlienLoader sample (SHA256: e739217419f83cf7351c18094d5147cf0183bdcee4271b06a75b8b4f7b38766c).

Pikabot

To get to the final stage – Pikabot – we need to detonate the whole attack chain in our pipeline in an automated way, which can be difficult. For example, to automatically open the ZIP and run the embedded Java file, the password must be extracted from the email or cracked. Further, to download and execute the Pikabot payload, it needs to be online and available to access, not blocked by geofencing techniques. It's possible to write static and dynamic detections for the Pikabot malware and create network signatures on its command-and-control (C2) traffic. It is very useful for defenders and analysts if a sandbox can extract the configuration of the Pikabot malware to support further hunting and research, as well as incident response of infected machines.

Many things can disrupt automated detection and forensics extraction, and that's why it's even more important to have multiple layers of detections in place and a solid defence in depth concept.

MITRE ATT&CK

By successfully detonating the entire attack chain, researchers were able to classify the individual elements at every stage, including the malware loader (AlienLoader) and the final payload (Pikabot) with its malware configuration extracted. This can inform campaign creation, including forensics and IOCs. By focusing on TTPs and not on atomic IOCs we can also produce a *MITRE ATT&CK* matrix from our dynamic sandbox detections. This can be useful to compare against existing security protection mechanisms to ensure an organization is protected against a certain technique. The following *MITRE ATT&CK* matrix focuses again on the Java dropper AlienLoader and its delivery of the final payload Pikabot. This should emphasize that protection against the delivery of malware is key to preventing the detonation of the final payload.

Matrix for our AlienLoader sample:

1. Email with encrypted ZIP attachment sent to victim answering an existing email communication from a compromised account.

   Resource development

   - T1586.002 Compromise Accounts: Email Accounts

   Initial access

   - T1566.001 Phishing – Spearphishing Attachment
   - T1199 Trusted Relationship – Thread Hijacking

2. Victim opens ZIP archive with password provided in the email and double clicks on the Java .jar file to execute it.

   Execution

   - T1204.002 User Execution – Malicious File

3. Pikabot payload gets downloaded via HTTPS from the C2 by AlienLoader and a new hidden window is created to start the downloaded payload in a new process after a sleeping time.

   Defence evasion

   - T1497.003 Virtualization/Sandbox Evasion – Time Based Evasion
   - T1564 Hide Artefacts – Hidden Window

   Command and control

   - T1071.001 Application Layer Protocol – Web Protocols

TA402 (APT)

Another example highlighting how DDX and intel can collaborate is TA402, also known as Molerats. In recent years, researchers have observed a significant shift in the threat landscape towards sandbox and detection evasion. This includes sophisticated geofencing (like limiting malicious payloads or next stages to a limited subset of requests of interest to the threat actor) and techniques detecting and blocking every kind of automation (like leveraging CAPTCHA checks, and blocking ASN ranges of known sandbox environments). Attackers often scrutinize browser requests and network artefacts to discern the presence of a sandbox environment.

A very basic, but effective, way to do this is to check the geolocation of an IP address visiting a URL. The threat actor can configure the server not to deliver the payload if the geolocation doesn't match a targeted region. With this technique, automated detection pipelines that use an exit gateway that's not within the target region are not going to see the payload and will likely return a benign verdict for their scan.

TA402 [7] is a likely Palestinian-aligned advanced persistent threat actor. TA402 typically restricts payload URLs [8] to deliver malware only if the geolocation fits the Middle East region. If not, they redirect the user who clicked the URL to a legitimate news website.

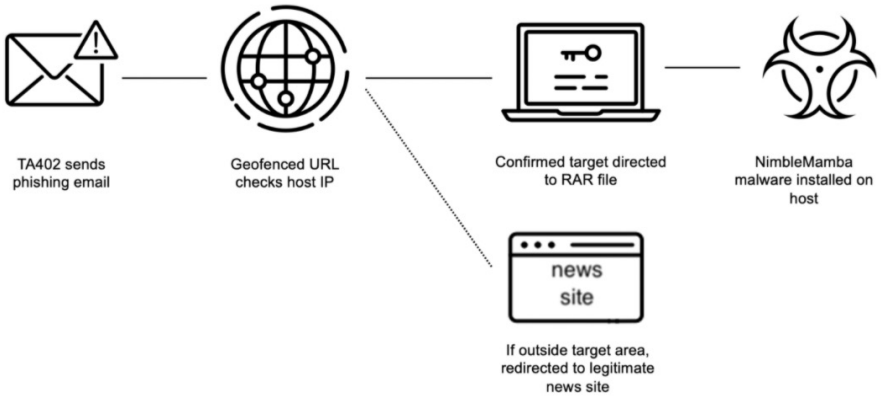

Figure 7: Example historic geofenced payload delivery by TA402.

Looking at the DDX concept, the pyramid of pain, and defence in depth, there are multiple detection opportunities to identify the email and URL as malicious, even though detonation might be interrupted in the pipeline due to geofencing. Often, the redirects used in geofencing techniques are unique enough to condemn because such behaviour is not normal, even though no real malicious behaviour has been observed. In any case, detecting the geofencing technique can be used to trigger a new scan of the URL with a different gateway IP address that is able to download the payload and then detonate, detect, and extract configuration from the malware. So even though the payload delivery is protected, it is still possible to automate the detonation with a second scan.

Of course this can be expanded not only to network evasion techniques, but also keyboard checks, computer language checks, and other kind of host artefacts. If the malware demands a certain value to fully reveal its behaviour and detonate, it's possible to spoof the value to the malware. Again, it's then possible to detect the evasion technique itself as well as the final malware behaviour that detonates. This is defence in depth, pyramid of pain, and DDX in action.

MITRE ATT&CK

Matrix for our TA402 campaign:

1. Email with geofenced URL that checks the host IP address when browsing to it and redirects to a decoy legitimate news site if outside target or a RAR archive containing the malware payload if inside target.

   Initial access

   - T1566.002 Phishing – Spearphishing Link

   Execution

   - T1204.001 User Execution – Malicious Link

   Defence evasion

   - T1614 System Location Discovery
   - T1036 Masquerading

INCREASING DETECTION CHALLENGES

The evolution of the cybercriminal threat landscape continues to pose new challenges for defenders. Advanced cybercriminal threat actors now incorporate rapidly evolving and novel attack chains, including using zero-day and n-day vulnerabilities.

Prior to 2022, macro-enabled documents were the go-to technique for threat actors distributing malware. Since mid-2022, when *Microsoft* began to block XL4 and VBA macros by default for *Office* users, researchers have observed widespread experimentation across the threat landscape. Threat actors continue to test various threat behaviours to determine the most effective method of gaining initial access via email. There is no reliable, consistent technique adopted by the entire threat landscape.

What this means is that there is something of a co-evolutionary relationship between threat actors and defenders. When new behaviours are identified, security teams must create new rules, detections, and tools to improve defences and detections. Once a technique becomes well-known, it becomes less effective, thereby forcing threat actors to try something different. It is a constant battle between threat actors and defenders.

To illustrate the dynamic nature of cybercriminal threats, we can once again look at TA577. Figure 8 shows an example of the number of unique TTPs observed in TA577 campaigns from January 2022 to May 2024.

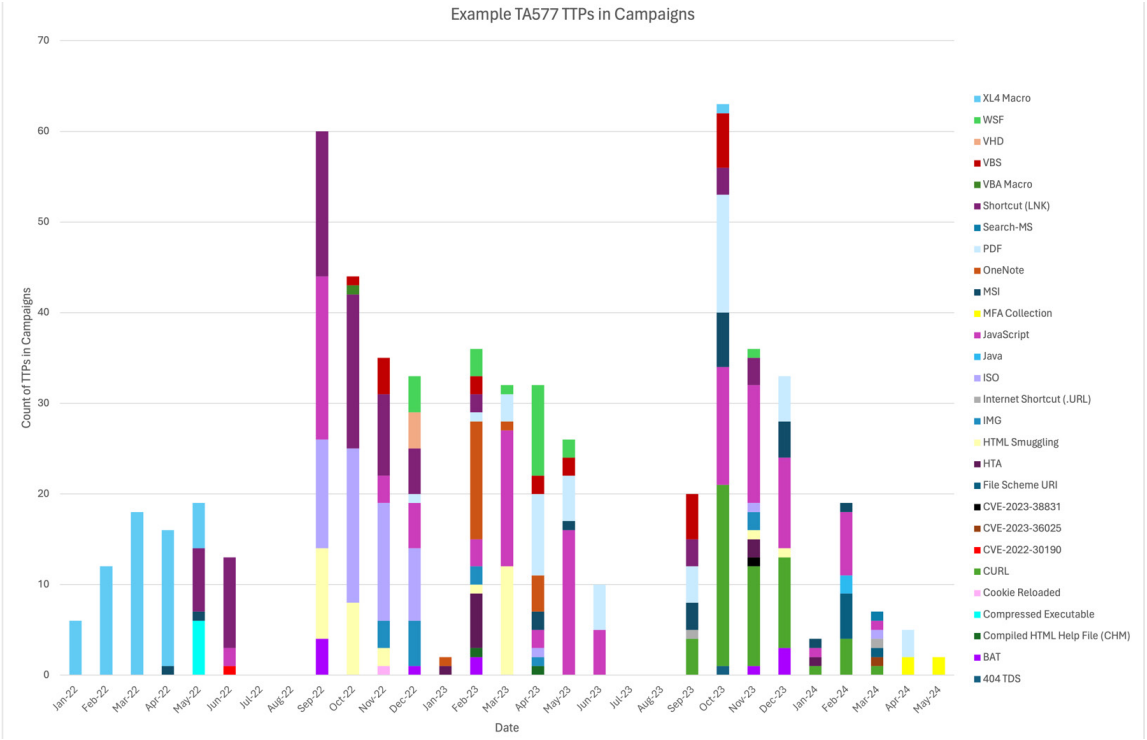

Figure 8: Example TTPs observed in TA577 campaigns.

Prior to 2022, TA577 almost exclusively used macro-enabled documents for malware delivery. Like many threat actors, TA577's behaviour changed significantly in mid-2022, and the graph shows how the actor went through waves of experimentation – from compressed executables to bypass mark-of-the-web attributes, to PDFs leading to internet shortcut (.URL) files or JavaScript payloads, to Java downloaders – in malware delivery and credential harvesting. (*Analyst note: prior to the end of February 2024, all TA577 activity was related to malware delivery. The actor switched TTPs from at least March – May 2024 to focus on credential phishing.*)

By identifying TA577 campaign changes as they happen, threat intelligence teams can provide new samples to detection engineering teams, who can create new signatures and defences based on behavioural characteristics. The sooner those are incorporated into the pipeline, the faster TA577's technique changes become less effective.

TA577 is not unique. Many financially motivated cybercrime actors, especially those involved in initial access for ransomware, have adopted new TTPs and attack development cycles. The changing landscape, with some actors demonstrating new attack chains daily, shows the stark necessity of threat intelligence-informed detection and defence.

BEST PRACTICES

Organizations sometimes silo detection engineering and threat intelligence functions, but the most effective defence happens with open lines of communication and collaboration between these groups.

Detection teams rely on threat intelligence groups to monitor the landscape, identify trends, and find gaps in coverage. Threat intelligence teams rely on detection engineers to fill those gaps, and in many ways, be an antagonist to threat actors. Threat-informed defence goes beyond blocklisting and IOC ingestion, but generating detections, intelligence, and rulesets based on real-world threats. The following are some general best practice suggestions for organizations to adopt:

- Build a foundation of mutual respect and relationships between detection engineering and CTI functions. Providing internal trainings, lunch and learn sessions, temporarily embedding employees in an opposite function, etc. can enable understanding and context for people to do their jobs more effectively.
- Leverage metrics to prioritize taskings. There are never enough hours in the day to look at everything, and taking some time to capture metrics and review the efficacy of taskings can help prioritize future work. For example, look back at old rules that had high volumes and high efficacy versus rules that took the same amount of time and were highly effective, but rarely fired. Identifying where an organization is seeing the best return on investment (ROI) of time, resources and skills can help scope tasks.
- Do not be exclusively reactive to what the threat actors are doing. While it is very important to leverage the landscape to drive security protections, it's also worth spending time to conduct purple teaming exercises and research to try to proactively identify what the next major threats and TTPs will be.

REFERENCES

[1] Bianco, D. The Pyramid of Pain. SANS. https://www.sans.org/tools/the-pyramid-of-pain/.

[2] McGuiness, T. Defense in Depth. SANS. 11 November 2001. https://www.sans.org/white-papers/525/.

[3] Larson, S.; Wise, J. Crime Finds a Way: The Evolution and Experimentation of the Cybercrime Ecosystem. Proofpoint. 12 May 2023. https://www.proofpoint.com/us/blog/threat-insight/crime-finds-way-evolution-and-experimentation-cybercrime-ecosystem.

[4] Office of Public Affairs U.S. Department of Justice. Qakbot Malware Disrupted in International Cyber Takedown. 29 August 2023. https://www.justice.gov/opa/pr/qakbot-malware-disrupted-international-cyber-takedown.

[5] MITRE ATT&CK. Compromise Accounts: Email Accounts. https://attack.mitre.org/techniques/T1586/002/.

[6] https://www.virustotal.com/gui/file/e739217419f83cf7351c18094d5147cf0183bdcee4271b06a75b8b4f7b38766c.

[7] Miller, J. TA402 Uses Complex IronWind Infection Chains to Target Middle East-Based Government Entities. Proofpoint. 14 November 2023. https://www.proofpoint.com/us/blog/threat-insight/ta402-uses-complex-ironwind-infection-chains-target-middle-east-based-government.

[8] Klinger, K.; Miller, J.; Mladenov, G. Ugg Boots 4 Sale: A Tale of Palestinian-Aligned Espionage. Proofpoint. 8 February 2022. https://www.proofpoint.com/us/blog/threat-insight/ugg-boots-4-sale-tale-palestinian-aligned-espionage.