VECTRA
SECURITY THAT THINKS

WHITE PAPER

How to detect malicious covert communications in today's encrypted network traffic

TABLE OF CONTENTS

Vectra® protects business by detecting and stopping cyberattacks.

As a leader in network detection and response (NDR), Vectra® AI protects your data, systems and infrastructure. Vectra AI enables your SOC team to quickly discover and respond to would-be attackers —before they act.

Vectra AI rapidly identifies suspicious behavior and activity on your extended network, whether on-premises or in the cloud. Vectra will find it, flag it, and alert security personnel so they can respond immediately.

Vectra AI is *Security that thinks*®. It uses artificial intelligence to improve detection and response over time, eliminating false positives so you can focus on real threats.

Cognito Detect empowers security teams to automatically pinpoint active cyberattacks as they're happening and quickly prevent or mitigate loss.

HIGHLIGHTS

- By mathematically analyzing the subtle patterns within network traffic, AI-powered Cognito Detect exposes the true underlying behavior, such as malware receiving command-and-control instructions, attackers using remote access tools, or attackers delivering malware updates.
- To expose hidden tunnels, Cognito Detect performs a highly sophisticated analysis of the traffic to reveal subtle abnormalities within a protocol that give away the presence of a hidden tunnel.
- Cognito Detect uses data science and packet-level machine learning to reveal the presence of external remote access without dependence on signatures.
- Through a careful use of AI and data science, Cognito Detect can separate software that masquerades as a browser used by humans and reveal the presence of hidden communications within allowed applications.

Introduction

Modern cyberattackers are patient, strategic operators who infiltrate and stealthily persist within a network over an extended period of time. These long-term attacks rely on ongoing communication to manage and coordinate the various phases of attack.

But unlike an exploit or malware infection – which only needs to succeed once to be effective – this command-and-control traffic must continually traverse the network perimeter without detection. As a result, attackers spend considerable time and effort to ensure their communications remain concealed.

Of course, command-and-control traffic is the means to a malicious end that typically involves the theft of data and assets. This exfiltration of data is the ultimate goal of an attack, attack, as seen even in ransomware,and requires attackers to move large amounts of data out of the network undetected. Covert exfiltration communications are invaluable to attackers and are the most damaging to victim organizations.

Attackers spend considerable time and effort to ensure their communications remain concealed.

It's critical to understand how attackers use covert communications across the network perimeter and how security teams can proactively spot them using Cognito Detect™, a cornerstone of the AI-powered Cognito® cyberattack-detection and threat-hunting platform from Vectra®. Understanding these techniques gives security teams the ability to rob attackers of the persistence and coordination that make modern attacks so successful.

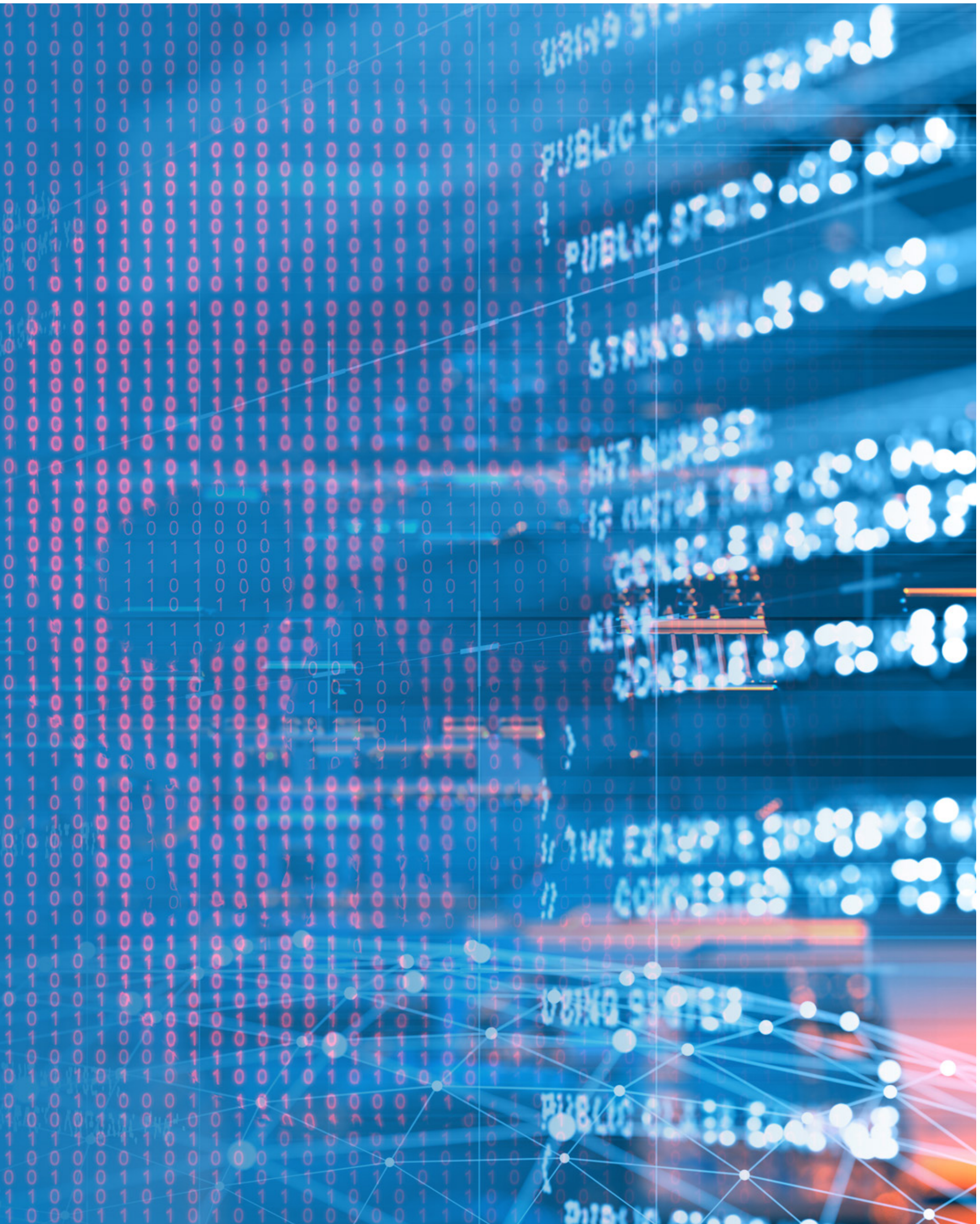
Uses of covert communications

The use of covert communications breaks down into two categories: Command-and-control traffic and exfiltration traffic. Both categories are vitally important to the success of an attack.

Command-and-control traffic broadly refers to any communication that allows attackers to remotely coordinate the attack. These communications can be fully automated, such as malware that checks in and gets instructions from a remote server. They can also be driven by an attacker that uses remote access tools to control an infected laptop.

Whether they are automated or driven by a human, command- and-control requires a bidirectional flow of information into and out of the network. This communication acts as the central nervous system of an attack, enabling a remote attacker to observe and adapt over time.

Once an attacker has access to key data or assets within a network, the focus shifts to smuggling those assets out. This is the goal of the exfiltration phase of attack. Unlike command-and- control, the exfiltration phase is more one-sided, with data flowing out of the network. Exfiltration also requires moving much larger amounts of data compared to command-and-control.



Why covert communications are difficult to detect

Covert communications only occur after an attacker gains some level of access to the network. Getting into the network proves that attackers can bypass the prevention-based security controls that are supposed to keep intruders out. This requires an understanding of how to evade IDPS, anti-malware, URL filtering, malware sandboxes and other signature-reliant technologies.

These attackers are likewise well-aware of the common techniques used to detect command-and-control traffic. Simply put, attackers who are skilled at obscuring their initial intrusions into the network are also likely to be quite adept at hiding control traffic.

Once inside the network, a world of evasion opportunities is opened to the attacker. Attackers now have a position of trust within the network and with that comes the opportunity to blend in with normal, trusted user traffic.

Attackers skilled at obscuring their initial intrusions into the network are very likely to be skilled at hiding their control traffic.

In addition to this position of trust, attackers control both ends of communication – the internal infected host and the external device or server. This introduces greater flexibility for attackers to use or modify permitted applications, encrypt their communications, and embed messages into seemingly normal traffic and content.

With this level of control, evading signatures or reputation lists requires an element of creativity on the part of attackers. But while the evasion opportunities are practically unlimited, the fundamental goals and behaviors of these communication channels are consistent and observable. Focusing on these fundamentals is the key to proactively finding and stopping covert communications.

Encryption

Encryption is designed to protect communications and keep out the prying eyes of those who try to snoop-in on a conversation. Unfortunately, these techniques work equally well for good guys and bad guys. Attackers have a variety of ways to encrypt and hide their traffic – from using standard SSL/TLS to using entirely customized schemes that are tailored to a specific operation or piece of malware.

The rise of HTTPS by default

Web-enabled applications based on HTTP have become standard in virtually every type of computing. Cloud-based applications like Gmail, software-as-a-service like Salesforce.com, and omnipresent forms of social media are increasingly accessed via a browser or built from Web technologies. This application shift has been accompanied by the use of SSL/TLS by default, thereby encrypting an ever-increasing amount of HTTP traffic into HTTPS.

While the use of encryption provides an obvious layer of protection for end-user sessions, it also has the effect of obscuring traffic from many network-based security solutions. This fact has not gone unnoticed by attackers, who have begun to use some of these same techniques to encrypt command-and-control or data exfiltration channels.

While the use of encryption provides an obvious layer of protection for end-user sessions, it also has the effect of obscuring traffic from many network-based security solutions.

Although some organizations use man-in-the-middle techniques and endpoint agents to decrypt outbound traffic for inspection, this approach has not been a cure-all. First, decryption is a resource-intensive process that introduces significant performance penalties. Anything that leads to slower performance is usually taboo within the broader IT organization.

Decryption additionally raises a variety of privacy concerns based on the content and geography of the user's connection. For example, financial and healthcare data are protected by regulatory mandates and some countries have strict privacy laws that prohibit inspection of encrypted user traffic.

Perhaps the most difficult challenge to SSL inspection involves the use of certificate pinning. Certificate pinning came about as a response to attackers who silently man-in-the-middle a victim's Web sessions using valid private keys that are capable of issuing new certificates.

Many online services have implemented tougher enforcement controls that require specific root certificates to be present – rather than any trusted certificate authority – to validate a connection. For example, a Google service may choose to trust only certificates from a specific trusted root certificate authority instead of any recognized certificate authority.

While this can thwart attackers who have stolen valid certificates, it can also break man-in-the-middle decryption methods used by security teams. This is another example in which making traffic more secure can also make it more difficult to directly inspect.

Custom encryption

While the use of SSL/TLS is a challenge, attackers are not limited to using standard off-the-shelf approaches to encryption. As noted earlier, attackers control both ends of a connection and are free to craft their own custom encryption strategies. This can be done by modifying existing protocols and strategies or devising their own.

This type of custom encryption can be very difficult to detect because the protocol might be unidentifiable and use any available port. Even if the traffic is detected, it would be nearly impossible to decrypt.

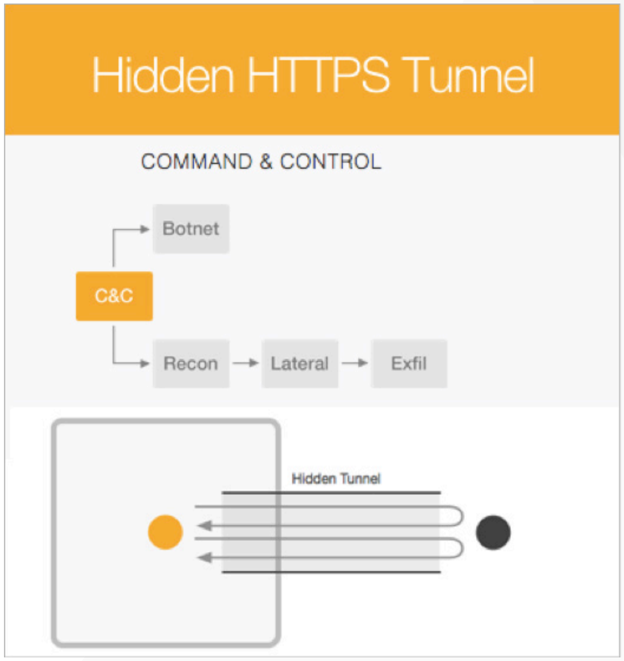
In some cases, simple obfuscation using custom encoding is all that is required. Either way, only the attacker knows how the data is encoded or encrypted, allowing it to bypass security without true inspection.

Vectra has developed a revolutionary new approach to packet-level network traffic to reveal its true function, even when traffic is encrypted.

How Cognito Detect reveals encrypted threats without decryption

To deal with encrypted threats, security must fundamentally be able to detect threats without inspecting the payload. This requires a new approach to network security. Transforming cybersecurity, Vectra has developed a revolutionary new approach that applies AI and data science directly to packet-level network traffic to reveal its true function, even when traffic is encrypted.

By mathematically analyzing the subtle patterns within network traffic, AI-powered Cognito Detect exposes the true underlying behavior, such as malware receiving command-and-control instructions, attackers using remote access tools, or attackers delivering malware updates.



Cognito Detect leverages data science and machine learning to reveal underlying attack behaviors, even when traffic is encrypted.

These patterns can be identified across application types, regardless of encryption. By leveraging AI and data science, Cognito Detect is able to detect threats that would be invisible to other solutions.

Cognito Detect is fundamentally different from other types of data science that simply re-analyze log or NetFlow data. These analysis techniques only examine secondary sources of data in an attempt to identify interesting correlations that might have gone unnoticed.

Secondary methods of analysis are also limited by the capabilities and granularity of the devices that generate the logs. If a threat is invisible to the original device that generates the logs, the event will likely remain invisible during the correlation phase.

By directly applying AI and data science at the packet level instead of the log level, Cognito Detect is able to identify threats inside encrypted channels that are invisible to signature-based systems as well as log and flow analytics solutions.

Hidden tunnels

Hidden tunnels are powerful tools that sophisticated attackers use to carry out command-and-control and exfiltration behaviors. These hidden tunnels are very difficult to detect because attackers will hide communications within multiple connections that use normal, commonly-allowed protocols.

For example, a seemingly normal HTTP-GET might carry a hidden malware request embedded within a text field. Likewise, the corresponding HTTP response may include instructions from the command-and-control server that are also hidden within a predetermined field.

This is not just limited to simple text fields. Covert communications can be embedded in a variety of fields as well as headers and cookies. Once again, the range of possibilities is limited only by the creativity of the attacker.

These techniques amount to a form of steganography within an allowed protocol, making detection very difficult. Even progressive decoding of the protocol is unlikely to reveal the communications because the messages are embedded within the allowed protocol. Since there is no internal protocol to decode, the traffic will appear normal.

How Cognito Detect finds hidden tunnels

To expose hidden tunnels, Cognito Detect performs a highly sophisticated analysis of the traffic to reveal subtle abnormalities within a protocol that give away the presence of a hidden tunnel.

Even though messages are carried within an allowed protocol, the concealed communications that make up the hidden tunnel introduce subtle anomalies into the overall flow of the conversation. These abnormalities include slight delays or abnormal patterns in requests and responses.

Using mathematical models, Cognito Detect identifies the presence of hidden tunnels within HTTP, HTTPS and DNS traffic. As noted previously, Cognito Detect even detects hidden tunnels within HTTPS without decrypting any traffic.

Cognito Detect also reveals hidden tunnels no matter how they are implemented. It doesn't matter what field attackers use to embed communications or whether they use a never-before- seen obfuscation technique. The variance from normal protocol behavior will expose the presence of the tunnel.

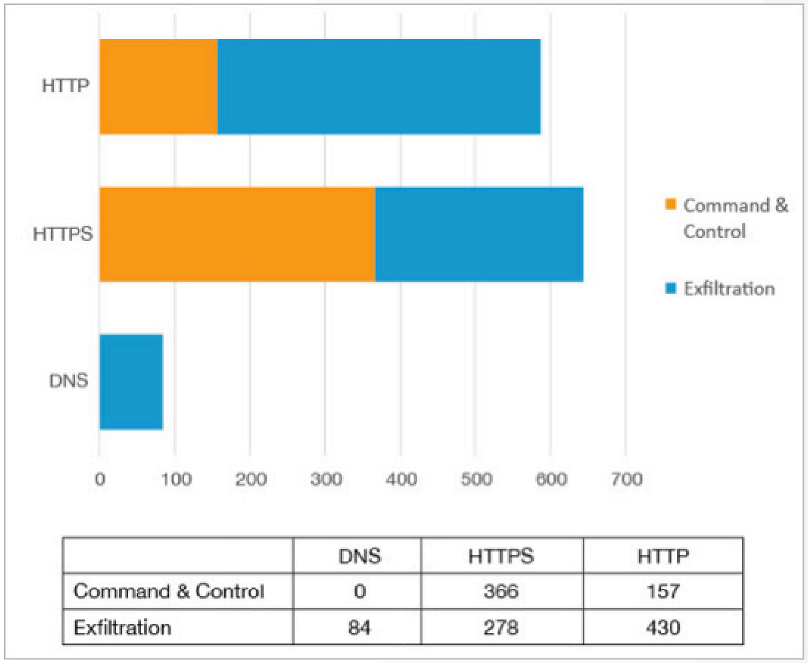
Hidden tunnels: A view from the front lines

Vectra researchers recently published the Post-Intrusion Report, which aggregates and analyzes real-world detection data from Vectra customer and prospect deployments. The study reveals that the use of hidden tunnels is on the rise and provides insight into the protocols that are preferred by attackers who construct hidden tunnels.

The study shows that HTTPS was the most popular protocol used for hidden tunnels – even more so than HTTP. This indicates that attackers will employ a variety of techniques to make sure their communications remain concealed.

The preference for HTTPS was strongest within hidden tunnels used for command-and-control, with HTTPS outpacing HTTP by a margin of 2-to-1. This runs counter to what would be seen in normal traffic. On average, HTTP is more than twice as common as HTTPS.

By hiding within encrypted traffic, attackers can avoid payload- based inspection altogether, and limit security teams to simplistic detection methods such as IP and URL reputation lists that are easy to evade.



	DNS	HTTPS	HTTP
Command & Control	0	366	157
Exfiltration	84	278	430

Real-world analysis reveals that HTTPS is the most common channel for hidden tunnels.

External remote access

External remote access tools give human attackers total hands-on control over infected devices, and are indispensable to launching targeted attacks. As an attack becomes more complex, it often becomes impractical or impossible to complete the attack using automated malware alone.

Instead, attackers must take manual control over attacks and use real-time human creativity to investigate and spread within the network. Remote access tools, also known as remote administration tools (RATs), are essential to this phase of controlling targeted attacks from the outside.

These vital attack tools are widely used and extremely difficult to detect. Signatures exist for the most common RATs. However, sophisticated attackers can easily customize their own RATs to avoid detection or devise their own using common remote desktop tools like the Remote Desktop Protocol, Virtual Network Computing or WebEx.

RATs are encrypted to conceal communications and will thwart attempts at payload analysis. They also make their initial connections inside a network and then hand-off control to remote attackers. In logs or via NetFlow analysis, this behavior looks like an ordinary web connection so attackers can blend in with normal user traffic.

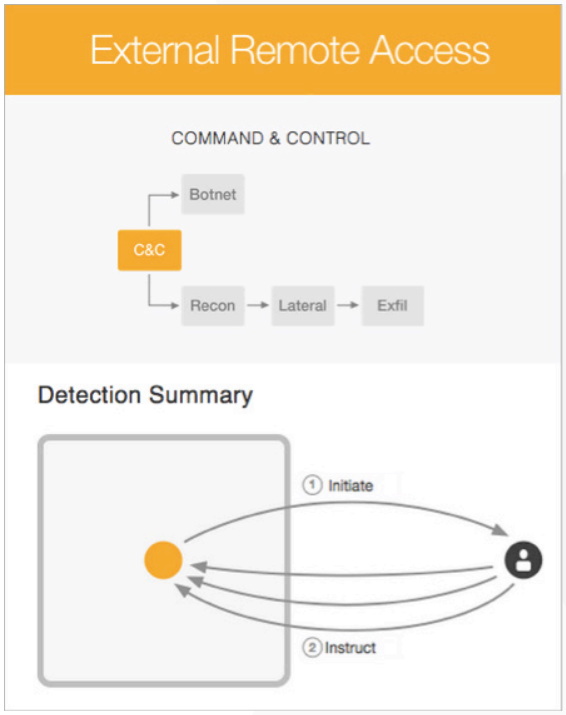
How Cognito Detect exposes external remote access

Cognito Detect uses data science and packet-level machine learning to reveal the presence of external remote access without dependence on signatures. While Cognito Detect employs multiple techniques, let's take a look at one of these methods as an example.

External remote access connections initiate from inside the network so they can look like normal user traffic. However, Cognito Detect performs a much closer analysis of these connections, including an analysis of the unique pauses within an open connection.

With external remote access, Cognito Detect identifies that the external entity breaks almost all pauses in the conversation. This provides a simple yet powerful indicator that the conversation is being controlled by an outside party. Consequently, Cognito Detect can identify external remote access generically, even malicious RATs that are customized or previously unknown to the security industry.

Cognito Detect also provides the crucial context that shows how external remote access is being used as part of a larger attack. When remote attackers take direct control over devices, they often do so to advance the attack.



Cognito Detect analyzes the patterns within live traffic to identify external remote access based on its unique behavior.

By correlating the presence of external remote access with reconnaissance scans, lateral movement and other malicious behaviors, Cognito Detect automatically prioritizes specific hosts at the center of an attack.

Cognito Detect also enables security teams to mark proprietary databases, credit card databases, medical records and other critical assets, and instantly view external remote access in the context of these assets. While administrators may use remote desktop tools from time to time, they should rarely have a direct connection to critical assets and systems from the Internet.

External remote access: A view from the front lines

Researchers at Kaspersky Lab recently provided a detailed analysis of a very successful criminal campaign dubbed the Carbanak APT. In this series of attacks, criminals infiltrated and stole over $1 billion from more than 100 different banks across the world.

The attackers used a variety of customized software, but were ultimately successful because they used common remote desktop tools to connect to internal systems that control the amount of funds in an account and the transfer of funds.

In other cases, attackers used remote desktop tools to manipulate systems that managed cash machines, causing them to dispense currency on demand. This is a prime example of why it's critical to identify any external remote access, including the use of approved applications. It ensures that critical systems are not being accessed by remote attackers.

Hiding within allowed applications

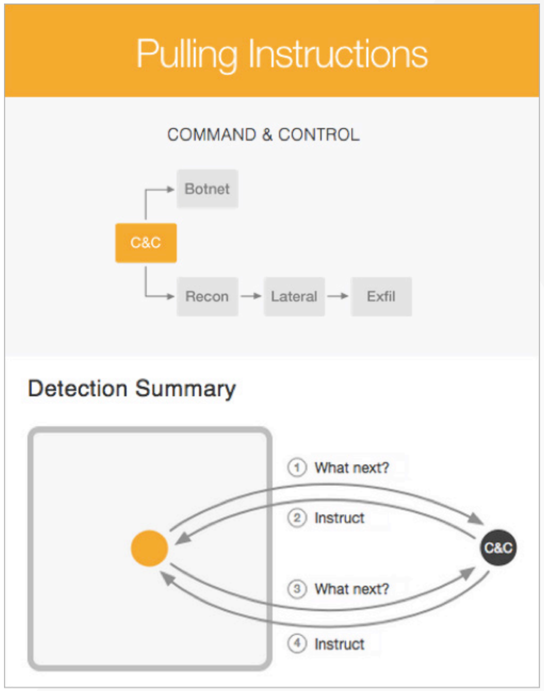
In busy networks, attackers love to hide in plain sight, and they do it by concealing communications within allowed applications or by emulating allowed applications. Although attackers can choose virtually any application, web traffic is their preferred hiding place because it is the most common type of traffic.

The vast amount of web traffic in a typical enterprise network provides an ideal backdrop for attackers who are hoping to blend in, and the endless variants of web-enabled applications offer a variety of ways to communicate.

One of the basic strategies used by attackers is to simply emulate a web-browser to blend in with the sea of enterprise web traffic. This technique is often used by malware to surreptitiously communicate with the outside world.

Some variations of this strategy use the HTTP POST method to communicate with a remote command-and-control server. Other more sophisticated strategies involve the use of a fully-automated browser and web session to send and receive instructions to the malware.

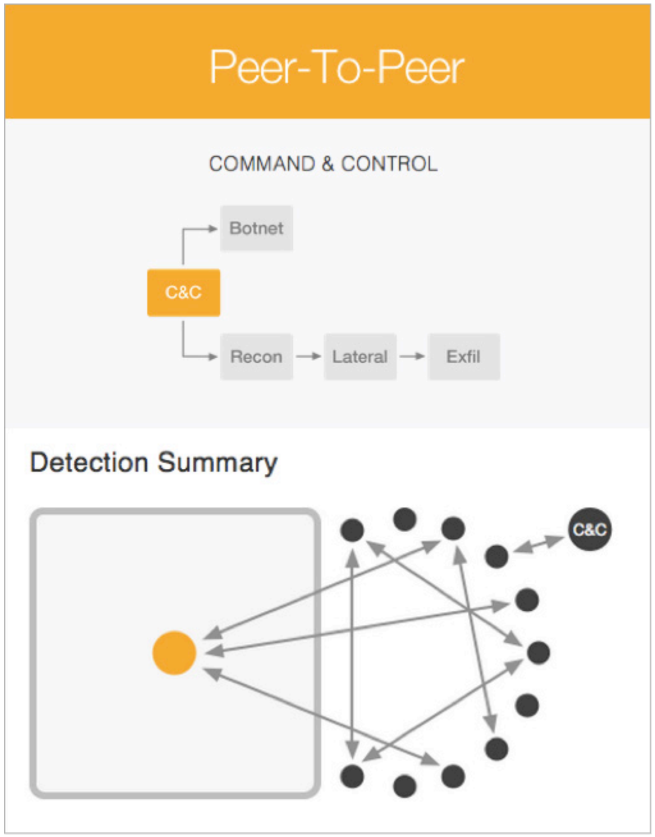
Although all these strategies attempt to blend in with normal user traffic, they share a steadfast and common trait – they are not traffic that has been generated as a result of human actions.



Cognito Detect analyzes conversations and identifies the unique patterns of command- and-control, regardless of the application used to carry the traffic.

How Cognito Detect finds hidden communications in allowed applications

Cognito Detect focuses on the aforementioned common denominator to identify malware that masquerades as a person. This can be accomplished in a number of ways. The first identifies unique patterns of command-and-control behavior, while the second identifies traffic that is recognizably the work of a machine and not that of a human.



In the first case, Vectra data scientists identify the unique patterns of various command-and-control behaviors. For example, command-and-control traffic is often used by malware to obtain new instructions from a command-and-control server.

This simple call and response creates an identifiable pattern of behavior that shows automated software on a machine inside the network is consistently performing rote actions and requesting new instructions from an outside source. This behavior is a sign of automation and is distinguishable from the actions of bona fide end users.

A scientific analysis of traffic also reveals a variety of behaviors that are distinctly not driven by human action. For example, while HTTP GETs and POSTs are common in real end-user sessions, malware makes use of them at rates that are impossible to achieve or highly regular intervals.

Through a careful use of AI and data science, Cognito Detect can separate software that masquerades as a browser used by humans and reveal the presence of hidden communications within these allowed applications.

Hiding in allowed applications: A view from the front lines

Researchers repeatedly identify advanced malware in the wild that automates applications to perform hidden communications. One example is malware that used Google Gmail as a command-and- control channel.

The malware opened a browser and suppressed the window so it was not visible to the user. It then automated the operation of the hidden browser and used Gmail to send encoded data and receive new instructions in an encoded Python script.

This malware was highly successful because it avoided signatures by using an allowed application and communicated with a trusted domain and IP range. While all external indicators of the application appeared benign, the fundamental communication pattern was identifiable as command-and-control when analyzed with always- learning behavioral models.

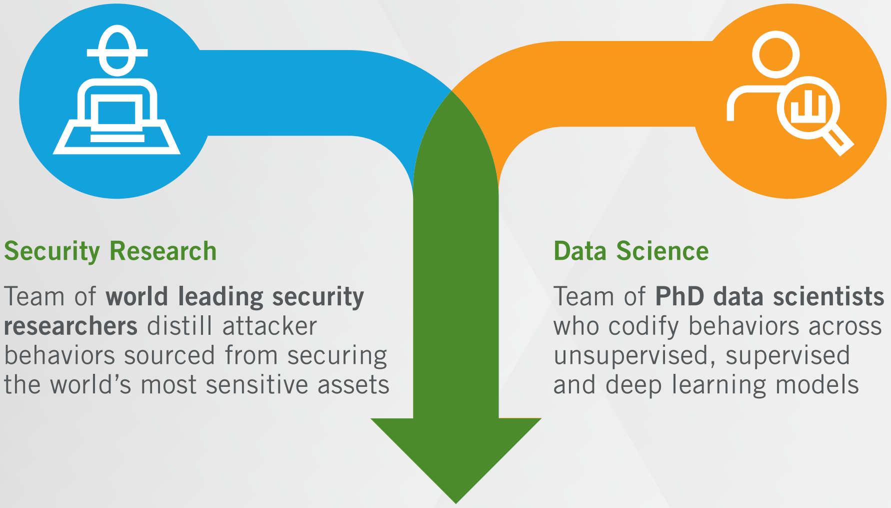
Conclusion

Covert communications are key enablers of cyberattacks that allow remote humans to patiently manage and direct their attacks undetected. Attackers choose these vehicles specifically for their ability to evade signatures, malware sandboxes and reputation lists.

Despite their ability to easily evade traditional security defenses, these covert channels are not invisible. Cognito Detect employs a unique approach that applies AI and data science directly to network traffic to reveal the true behavior and purpose of the traffic. This enables Cognito Detect to expose covert communications, regardless of the applications being used.

By identifying how these covert channels actually work, Cognito Detect empowers security teams to automatically pinpoint active cyberattacks as they're happening, correlate threats with the hosts that are under attack, prioritize attacks that pose the greatest business risk, and quickly prevent or mitigate loss.

Cognito Detect employs a unique approach that applies AI and data science directly to network traffic to reveal the true behavior and purpose of the traffic.



Security Research

Team of **world leading security researchers** distill attacker behaviors sourced from securing the world's most sensitive assets

Data Science

Team of **PhD data scientists** who codify behaviors across unsupervised, supervised and deep learning models

Security Analyst in Software

97% of the MITRE ATT&CK framework
Security **enrichments** (e.g. privilege)
Automate Tier-1 activities: **34X workload reduction**

For more information please contact a service representative at info@vectra.ai.

Email info@vectra.ai vectra.ai

Version: **092320**