Decoding the Encoded

Posted on November 26, 2024

Authors:
Ed Gibbs, Field CTO, WHOIS API Inc.
Jeff Vogelpohl

Introduction

Growing up, I remember the vast array of candies and ice cream flavors while visiting quaint candy shops. Today, we're overwhelmed by the plethora of technologies any imaginative person could want – thanks to the provocativeness of human ingenuity. As flavors were designed for these memories of delightful treats, this same ingenuity has brought technological advancements like AI to aid and improve all life whereas some provide just the opposite. Our adversaries continuously exploit and weaponize our ingenuity to degrade life. Life is worth protecting.

Throughout the world, people strive to uphold their duty to serve and support the effort to protect life. People run banks, businesses, churches, education, eateries, energy, entertainment, financial institutions, healthcare, hospitality, manufacturing, sanitation, tech, transportation, utilities, and anything with a bit of everything. Everyone has their role.

Back in September, I recalled Ed Gibbs' statement, "...the long lines of SMTP data that require a PhD to decipher…." Ed's statement holds true for many seasoned techies spending countless times sifting through code, data, and logs attempting to understand where it all went wrong. Regardless of time, we have educational resources available to learn so we can adapt to the ever-changing landscape we call technology, the Internet, and life as we know it – because the air we breathe at home is the same everywhere else. Our lives are seamlessly integrated with technology.

Identifying Our Data

The biggest question that’s on everyone’s mind is, “what data are our adversaries after?” The question is too broad, and the answer evolves drastically from any trade or state perspective. Protecting our trade and state secrets are constant regardless of how we identify the data. According to the Internet Crime Complaint Center (IC3) in 2023, reported financial losses were over $3.4 billion for people over 60 years old; all Internet crimes amounted to over $12.5 billion in reported losses. Life is always at stake with many facing an existential crisis. We should not lose hope.

As people play their part, our adversaries are antagonists in our data story line. Data could include code, contracts, databases, designs, documents, drawings, files, logs, maps, plans, notes, pictures, spreadsheets, statements, and a whole slew of information waiting to be analyzed by a trove of adversaries. The endgame is our loss – financially, politically, it all matters. The dark web is riddled with our data, and the data that’s been deleted remains in a deep dark area. Prying eyes always find just enough light to read.

Identifying data matters, but knowing life is always at stake is vital in our efforts to protect both.

Adversary Delivery Methods

Our adversaries deliver malicious payloads using any number of modern mediums (not ghostly mediums either) including but not limited to email, electronic storage devices, QR codes, SMS/text messages, and websites. What are malicious payloads anyway? Well, they are in fact malicious with the intent to deceive and perform nefarious actions such as encrypting data for ransom as with ransomware. Other forms include hijacking systems and stealing information.

Let’s keep it simple – malicious payload delivery methods include email, storage devices, QR codes, text messages, and websites. It’s important to realize that email is seamlessly integrated in our lives – personally and professionally. Sometimes people include both on a single device. Downloading apps, playing games, reading email, and catching up on stories online may ease stress – not realizing that any one of those actions could compromise a life, which is always at stake. Protecting email and all forms of communication are important.

Uncovering the Nefarious Content

Statistics aside, Cybersecurity greatly affects all facets of our lives. Adversaries threaten the integrity of commerce, education, elections, government, healthcare, telecommunications, and every other system we rely on to protect our way of life. They attack us from afar and from within – often hiding in plain sight or disguised in unexpected ways. Where’s nefarious content hiding anyway? It’s in apps, email, print, online – literally everywhere people are watching, listening, or feeling. People’s senses are overloaded with content and leaving them unsure of what’s good or bad – you get the idea.

Of the 33 million businesses across the United States as of 2024, could you identify the unique design elements in each brand's communications, such as emails, to determine if they are phishing attempts? The sheer scale of these numbers is staggering. What about web pages? What about phone calls? Lives are at stake.

Nefarious content doesn’t just exist in what we’re watching, listening to, or feeling – it’s also embedded in the payloads feeding our devices, driving them to execute malicious actions, whatever they may be. It happens in an instance.

You’re probably already thinking about the time you received the sus email just this morning, the text regarding the incorrect address for your delivery, or the fake tech support website you were reading about the error for your favorite fruit accounting system. Regardless, nefariousness is going on.

What’s going on in the wild? Nefariousness is making headlines. Feel free to tag #HelpStopNefariousness to expand reach because education is key. No matter what protections are in place, the last line of defense is us. Let’s take a closer look at some of the deceptive techniques adversaries use to deceive us and deliver their payloads.

- The use of *Homoglyphs* – characters that look nearly identical but aren’t, like "O" and "0" – is a common tactic to deceive and mislead. Another example is combining two characters, such as “rn,” to mimic a single character like “m” – the possibilities are endless.

- The use of what’s called “*Branded Sub-squatting*” – creating DNS subdomain records with branded names within a domain tree, such as 'my-brand-here.support.help-stop-nefariousness.tld' – is a tactic designed to display the brand name first, encouraging users to overlook the rest of the domain name. This technique is particularly effective in phishing, as users often focus on the email address prefix and only a limited portion of the domain name, making it a favorite tool for scammers.

- The use of *Unicode* characters to disrupt a user interface by creating a break mechanism that prevents programs, such as email clients or web browsers, from operating properly can pose a serious security risk, especially if the system becomes unresponsive.

- The use of *shared content* such as audio, emojis, fonts, GIFs, images, and videos hosted on suspicious cloud platforms.

- The use of *polyglot files* – crafted to appear as multiple file types simultaneously – allows attackers to bypass security measures, exploit vulnerabilities in file parsers, and deliver malicious payloads while evading detection. For example, a Word document embedded with a polyglot structure could masquerade as a legitimate .docx file while also containing a malicious ZIP archive, enabling attackers to execute hidden payloads when specific conditions are met.

- The use of *obfuscation techniques* resembles a 3D padlock with infinite combinations – for example in JavaScript, converting ANSI characters to decimals, using a Dec function to decode the array, and executing it with Eval. Modern techniques, such as Base64 encoding to embed and obfuscate content, the use of Unicode escape sequences like \xHH (hexadecimal) and \uHHHH (Unicode), HTML entities, and even encoding the encoded

\uHHHH character array itself to further hide their tactics, add layers of complexity. Combined with a wealth of similar methods, this ever-evolving approach makes detection and analysis almost impossible.

- Most importantly, *embedded files* within files can bypass security protections and data loss prevention (DLP) measures. The risk is even greater with encrypted files, as embedded content may go undetected. The risk also extends to email messages.

What do all these deceptive techniques mean? Maliciousness.

Decoding the Encoded

Finally, you can turn to the person next to you in the movie theater as if you just heard the movie's name for the first time – we're amazed, too!

By identifying data, exploring modern delivery methods, and gaining a bird's-eye view of the endless combinations of nefarious and obfuscation techniques, we can better anticipate what lies ahead – allowing us to break it all down through decoding, deciphering, and decrypting, ultimately uncovering maliciousness and preventing leaks or attacks.

How can we manage the workload of such a daunting task? By leveraging amazing tools like CyberChef and other great solutions, including WHOIS API's powerful technology designed to deliver exceptional cyber threat intelligence – combined with a willingness to learn.

CyberChef, often referred to as "The Cyber Swiss Army Knife," is a web-based tool designed to simplify complex data processing tasks. It provides a user-friendly interface for decoding,encoding, encryption, data analysis, and conversion of data formats. With its drag-and-dropfunctionality, users can create "recipes" to automate workflows and handle tasks like Base64decoding, hash generation, or file format transformation. CyberChef is an essential tool forcybersecurity professionals and analysts due to its versatility, accessibility, and powerfulfunctionality, making it a go-to solution for decoding, deciphering, and decrypting data quickly andefficiently.

WHOIS API, Inc. provides a comprehensive suite of tools designed to enhance cyber threat intelligence by offering detailed domain registration and ownership data. With its ability to query vast databases of WHOIS records, it helps cybersecurity professionals identify potential malicious actors by uncovering domain history, registrar information, and associated IP addresses. Its advanced capabilities, such as bulk domain lookups, reverse WHOIS searches, and domain monitoring, make it invaluable for tracking patterns in phishing campaigns and uncovering homoglyph attacks. By integrating WHOIS API into workflows, analysts can gain actionable insights to decode, decipher, and decrypt the origins and intent behind cyber threats, ensuring a more proactive defense against emerging attacks.

The integration of tools like CyberChef and WHOIS API demonstrates how automation can be a game-changer in protecting against data leaks and cyberattacks. These tools streamline complex processes that would otherwise require significant time and expertise, enabling cybersecurity professionals to act faster and more effectively.

- **Efficient Threat Analysis:** CyberChef’s ability to process and analyze vast amounts of data in real time allows security teams to decode, decipher, and decrypt potential threats quickly, identifying vulnerabilities before attackers can exploit them. Automated workflows created within CyberChef minimize human error and enhance response times.

- **Proactive Threat Intelligence:** WHOIS API automates the process of gathering and analyzing domain registration data, uncovering patterns in phishing campaigns, and detecting suspicious domain activities. By continuously monitoring domain registrations and ownership changes, organizations can preemptively block or investigate potential threats before they escalate.

- **A Unified Defense Strategy:** When combined, these tools exemplify the power of automation in cybersecurity. They not only reduce the workload on security teams but also enable a more proactive and predictive approach to defense. Automation ensures that repetitive tasks like data analysis, decryption, and domain monitoring are handled efficiently, freeing experts to focus on high-level decision-making and mitigation strategies.

In an era of increasingly sophisticated cyber threats, automation offers a scalable, reliable, and effective way to enhance security, reduce response times, and stay ahead of attackers.

Example: Analyzing Email Header and Body

To see how these tools work, let’s combine CyberChef and WHOIS API to decode and analyze a suspicious email message:

- Analyze the Email Header with CyberChef
    - Copy the email header from your email client and paste it into the CyberChef Input box.
    - Use the “Extract Domains, Extract Email Addresses, or Extract IP Addresses” recipes to decode and analyze the routing information. Utilize each recipe separately.
    - Identify any unusual domains, email addresses, or IP addresses that might indicate phishing. Watch out for false positives.
    - Copy or Save the Output for further analysis with WHOIS API.
- Decode Suspicious Text with CyberChef
    - If you find encoded text like Base64 in the email body, feed it into CyberChef.

  - Use the recipe “From Base64” to decode the encoded string.
  - Review the output for signs of malicious intent, such as URLs, payloads, or commands.

- Investigate Domains with WHOIS API
  - Extract any domains from the email header or links in the email.
  - Use WHOIS API to look up detailed registration data and history.

Identify if the domain is newly registered or associated with malicious activity.

Example: Investigating Suspicious Web Page Source Code

If you encounter a suspicious web page, its source code might contain hidden threats, such as obfuscated scripts, encoded payloads, or links to malicious domains. Using tools like CyberChef and WHOIS API, you can investigate its intent without diving too deep into manual code analysis. Here’s how:

Example Code (Hello World):

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<script>
(() => {
 const deca = [99, 111, 110, 115, 111, 108, 101, 46, 108, 111, 103, 40, 39, 72, 101, 108, 108, 111, 32

 const script = deca.map(code => String.fromCharCode(code)).join('');

 const site = "bWFsaWNpb3VzLXNpdGUudGxk";
 eval(script);
})();
</script>
</head>
</html>
```

- Extract the Web Page Source Code
    - Use the Example Code above or use a simple Python script to save the web page's source code as a file. Downloading potential malicious code is a risk.
    - Save the source code to a file.
- Analyze the Source Code with CyberChef
    - Open file as input in the CyberChef Input box.
    - Use recipes like “Extract URLs” to pull out all embedded links or “From Base64” to reveal hidden data.
    - Highlight anything unusual, such as links to domains that don’t match the page’s purpose or encoded content that needs further analysis.
- Decode Encoded Content with CyberChef
    - Add a new input tab, then copy the “site” variable from the Example Code and paste into the CyberChef Input box.
    - Use the recipe “From Base64” to decode the encoded string.
    - Carefully review the output.
-

Investigate Extracted Domains with WHOIS API

- Take any domains or URLs extracted from the source code and run them through WHOIS API.
- Review the results for details like domain registration dates, ownership, and associated IP addresses.
- Look for red flags, such as recently registered domains, privacy shields, or unusual domain patterns.

Why This Matters: This process allows you to dig deeper into the web page's intent without requiring extensive coding expertise. CyberChef simplifies data extraction and decoding and WHOIS API provides critical intelligence on suspicious domains. Together, these tools offer a streamlined, effective workflow for uncovering threats hidden in web page source code and protecting against potential attacks.

Final Thought

The art of cybersecurity is not reserved for experts. Any non-technical person with a willingness to learn can delve into this field to better understand and analyze the digital world we live in. With the right tools and curiosity, you can uncover hidden threats, protect yourself and others, and contribute to a safer online environment.

IOC Index

Indicator	Value	IOC Type
[104 116 116 112 115 58 47 47 49 50 55 46 48 46 48 46 49]	https://127.0.0.1	URL

[99, 111, 110, 115, 111, 108, 101, 46, 108, 111, 103, 40, 39, 72, 101, 108, 108, 111, 32, 87, 111, 114, 108, 100, 39, 41, 59]	console.log('Hello World');	JavaScript Code
bWFsaWNpb3VzLXNpdGUudGxk	malicious-site.tld	Domain Name
eval(script);	deca.map(code => String.fromCharCode(code)).join('')	JavaScript Code