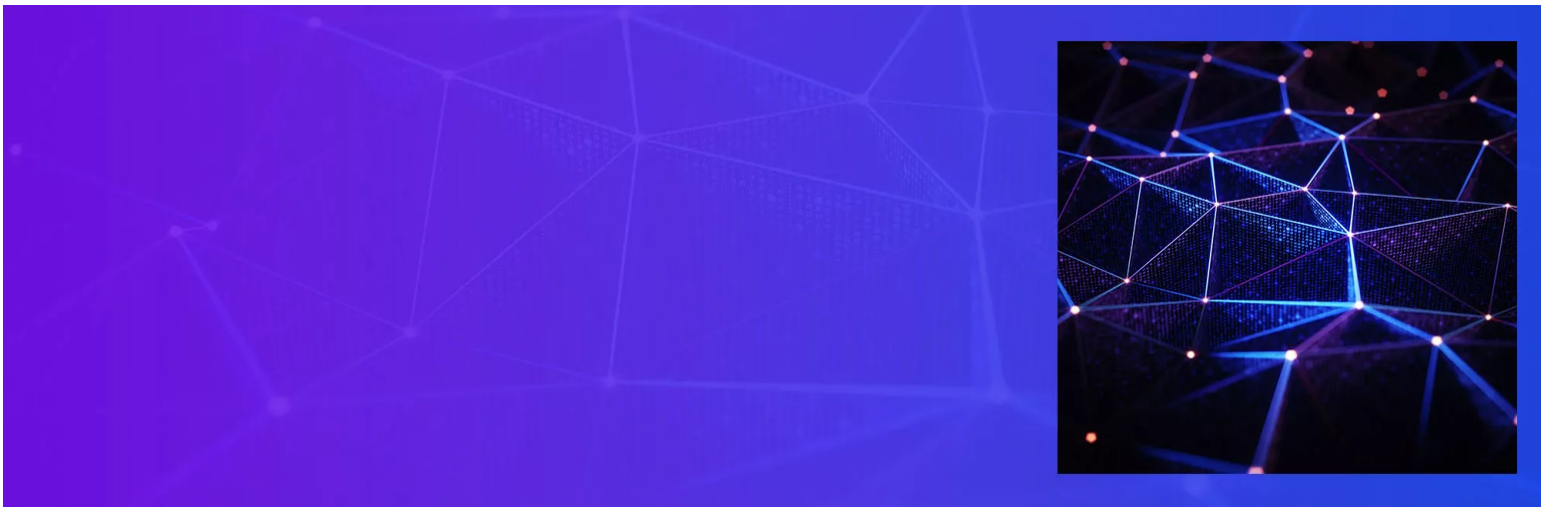


This article is for security and botnet enthusiasts, aiming to answer questions such as "What are the typical uses of botnets?" and "How can researchers build their own botnet for legitimate research purposes?" Uncover the practical applications of botnets and explore the methodology for constructing your own, all within an ethical and controlled framework.

Introduction

Cybercriminals are increasingly using botnets to launch devastating attacks, making them a major threat in today's digital landscape. Recent reports highlight the scale and sophistication of these attacks. For instance, Cloudflare's Q1 2024 data shows that 37% of all DDoS attacks were HTTP-based and frequently driven by botnets. Meanwhile, the emergence of advanced threats like the Jenkins Flood has surged by an astonishing 826% quarter-over-quarter.

The situation is further exacerbated by findings from CrowdStrike's 2024 Global Threat Report, which notes a 75% spike in cloud intrusions. Adversaries are utilizing botnets to execute stealthier and more effective attacks, leveraging these networks to exploit vulnerabilities across various sectors.

Given this alarming trend, understanding and countering botnets is crucial. In this blog post, we investigate what botnets are used for and guide you on how security researchers can construct their own for legitimate research purposes. By creating controlled botnets, researchers can dissect the mechanics of these malicious networks, gaining insights that are pivotal for developing stronger defenses.

It's essential to emphasize that our approach to building botnets is strictly for educational and research purposes, conducted within ethical boundaries and secure environments. This hands-on experience is invaluable for enhancing our understanding of botnet behavior and improving our cybersecurity strategies.

What are botnets?

A botnet is a network of computers infected by malware and controlled remotely by a command-and-control center (C2 servers or CnC). The concept of botnets dates to the early 2000s. In 2001, a worm called "MafiaBoy" infected thousands of computers, creating a network of compromised machines under remote control. This incident showcased the formidable power of botnets and their capacity to execute large-scale attacks. Over the years, botnets have grown significantly in sophistication and scale, continuing to pose a substantial threat to cybersecurity.

Use cases of botnets

In the realm of cybersecurity threats, botnets play a significant role in carrying out various malicious activities. These networks of compromised devices, controlled by a Command and Control Center (CnC), exhibit a range of offensive capabilities despite variations in their names and techniques. Here are a few scenarios that explore the threat surface and the roles of botnets, along with the corresponding attack vectors:

- **Launching DDoS Attacks**
  Botnets are notorious for orchestrating Distributed Denial of Service (DDoS) attacks. By leveraging a vast network of compromised devices, they can overwhelm Internet Service Providers (ISPs), websites, APIs, and other internet-connected components. This results in disrupting services and causing significant downtime.

  Attack Vectors: Botnets utilize various attack vectors such as amplification attacks, SYN floods, or DNS reflection attacks to generate massive volumes of traffic, exhausting the target's resources and causing service disruptions. One of the notable botnets is Mirai, which is known for leveraging IoT devices to launch powerful DDoS attacks.

- **Reputation Manipulation for Spam and Phishing campaigns**:
  Botnets are a critical tool in the arsenal of cybercriminals, particularly in manipulating the reputation of newly created malicious domains to bypass spam filters and send out large volumes of spam emails. By artificially boosting the reputation of these domains, botnets can evade detection mechanisms and deliver unwanted and potentially harmful messages to unsuspecting recipients.

  Attack Vectors: Botnets, through the coordinated efforts of compromised devices, engage in automated activities that generate artificial reputation scores. This tactic tricks anti-spam systems, significantly increasing the chances of successful email delivery. A notable example is the 3ve botnet, notorious for manipulating domain reputations to facilitate extensive spam and phishing campaigns.

  These sophisticated phishing campaigns often aim to steal sensitive information by luring recipients into clicking malicious links or opening harmful attachments. By understanding and mitigating these attack vectors, security professionals can better protect their networks from the pervasive threat of botnets.

- **Solving CAPTCHA Challenges**
  Botnets possess the sophisticated capability to solve weak CAPTCHA challenges on websites, effectively mimicking human behavior during login attempts. By bypassing these security measures, botnets can gain unauthorized access to user accounts. Once inside, they can carry out further malicious activities, such as data theft, account takeovers, and launching additional attacks, all while evading detection.

  Attack Vectors: Botnets employ sophisticated algorithms or machine learning techniques to analyze and solve CAPTCHA challenges, effectively deceiving systems that rely on these challenges as a defense mechanism. Most notable examples include Satori, Necurs and eve, utilized advanced algorithms or machine learning capabilities for this attack vector.

- **Credit Card Information Theft**
  Botnets are formidable tools in the arsenal of cybercriminals, capable of stealing credit card information from unsuspecting users. By infecting devices with keyloggers and other malicious software, botnets can capture and transmit sensitive data, including credit card numbers, CVV codes, and personal information. This stolen data is then used for fraudulent transactions, sold on the dark web, or utilized for identity theft, leading to significant financial and personal harm to the victims.

  The financial impact of such stolen data is substantial. For example, basic credit card details can sell for an average of $17.36 on the dark web. The price can increase significantly if the card includes additional information, such as a higher credit limit or a PIN, with cloned physical cards fetching up to $171 each (PrivacySharks) (Comparitech).

  Attack Vectors: Botnets leverage keyloggers or similar techniques to intercept and exfiltrate keystrokes, enabling the theft of credit card details and potentially leading to fraudulent transactions. One of the notable botnets is Trickbot Trojan, which is often involved in banking trojans and credential theft, including credit card information.

- **Unauthorized Network Access and Persistence**
  Some botnets are also used for targeting corporate networks to gain unauthorized access and establish persistence within compromised systems. These botnets exploit vulnerabilities in network infrastructure, operating systems, or software applications to infiltrate the target environment.

  Attack Vectors: Once inside the network, the New Sysrv Botnet utilizes advanced techniques to establish persistence,

ensuring its ongoing presence and control over the compromised systems. It deploys persistence attack payloads, such as rootkits, backdoors, or other stealthy malware, to evade detection and maintain a foothold within the network environment. One of the notable botnets is The New Sysrv Botnet which has the ability to gain unauthorized network access and establish persistence highlights the importance of robust network security measures, including regular patching, vulnerability management, strong authentication, and network segmentation.

- **Cryptojacking**
  Botnets can be used to mine cryptocurrencies using the processing power of compromised devices without the owner's consent. This drains the resources of infected systems, causing slower performance and increased electricity costs for the victims.

  Attack Vectors: Botnets deploy mining software on compromised devices, often using sophisticated methods to remain undetected while continuously generating cryptocurrency for the attackers. A notable example is the Smominru botnet, which has infected hundreds of thousands of devices to mine Monero, a privacy-focused cryptocurrency.

- **Ad Fraud**
  Botnets can generate fake clicks on online advertisements, defrauding advertisers and skewing analytics. This type of fraud not only causes financial losses but also affects the integrity of online advertising metrics.

  Attack Vectors: Botnets use compromised devices to simulate human behavior, clicking on ads and generating fraudulent traffic. The Methbot botnet, for example, was responsible for generating millions of dollars in fraudulent ad revenue by creating fake video views and clicks on advertisements.

The global financial impact of botnets is staggering, with cybercrime linked to botnets costing the global economy approximately $600 billion annually (Council on Foreign Relations). These covert networks pose significant threats to individuals, organizations, and the digital ecosystem through various attack vectors such as crypto miners, keyloggers, and other malicious tools. The critical need for comprehensive cybersecurity strategies to mitigate these threats effectively cannot be overstated.

Build your own botnet, for fun and research

As a next step, we are going to explore various botnet frameworks available and demonstrate how to use these botnets in controlled environments to test an organization’s readiness and resiliency. This approach not only helps security researchers and developers deepen their understanding of botnet operations but also enhances practical skills in mitigating such threats.

Creating a self-controlled botnet can serve multiple purposes. For instance, it allows you to perform DDoS Resiliency Exercises, where organizations can simulate attacks to measure their defenses against distributed denial-of-service (DDoS) attacks. These simulations help identify potential vulnerabilities and put compensatory controls in place to bolster infrastructure security.

Additionally, these exercises are valuable for monitoring and recording response times during simulated attacks. This practice enables IT and security teams to refine their incident response strategies, ensuring they are well-prepared for real-world scenarios. By testing in a controlled environment, organizations can enhance their cybersecurity posture and ensure that they are ready to face actual botnet threats effectively.

Engage further to uncover the best practices for building and utilizing your own botnet in a safe and ethical manner, ultimately strengthening your cybersecurity defenses and resilience.

Caveats of building your own botnet

Building a self-controlled botnet for research purposes comes with unique characteristics that distinguish it from malicious botnets. For instance, a self-built botnet does not involve compromising devices from various organizations or multiple households. Consequently, achieving a diverse set of IP addresses, which is common in malicious botnets, is much more challenging with an ethically built botnet.

In a typical malicious botnet, the bots are compromised devices from numerous sources, providing a wide range of IP addresses. This diversity helps attackers evade detection, as the malicious activities appear to originate from different sources, complicating efforts to trace them back to a single origin. The distributed nature of these botnets allows for more effective obfuscation of the

attackers' activities, making it harder for defenders to identify and block the malicious traffic.

Conversely, a self-built botnet, created for ethical or research purposes, operates under a controlled framework. The participating bots in an ethical botnet are typically owned and managed by the researcher or the organization conducting the study. This means the IP addresses associated with these bots are not diverse, usually originating from a single source or a limited set of controlled devices.

The limited diversity of IP addresses in a self-built botnet is intentional and serves specific research or ethical purposes. It allows researchers to study botnet behaviors, evaluate detection techniques, simulate attack scenarios, and develop defensive measures. By having a controlled set of devices, researchers can closely monitor and analyze the botnet's activities without risking harm or compromising other systems.

While a self-built botnet may not exhibit the same distributed characteristics as a malicious botnet, it provides valuable insights into botnet operations. Researchers can study communication patterns, propagation mechanisms, command and control protocols, and the impact of botnet activities on network resources. These insights are crucial for developing effective defensive strategies and enhancing overall cybersecurity measures.

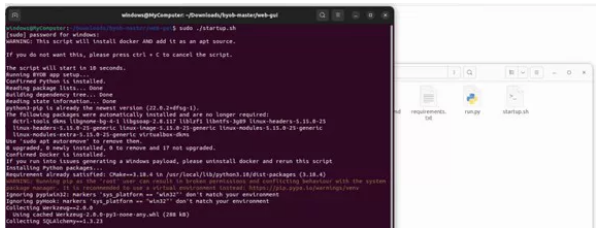
How to build your own botnet

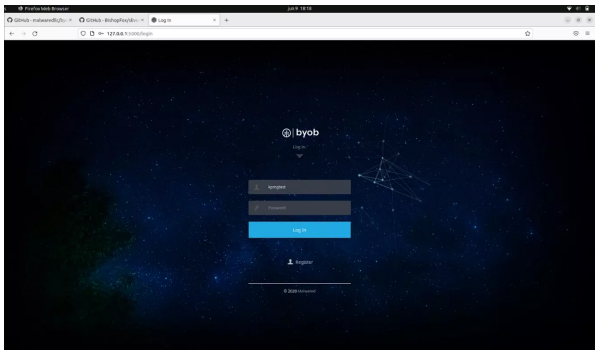
Okay, enough of the talking now. Let's try our hands on a real botnet framework. For this project, we have specifically chosen **BYOB – Bring your own Botnet,** which is an easy-to-use botnet framework and gives a glimpse of how easy these attacks and techniques have become.

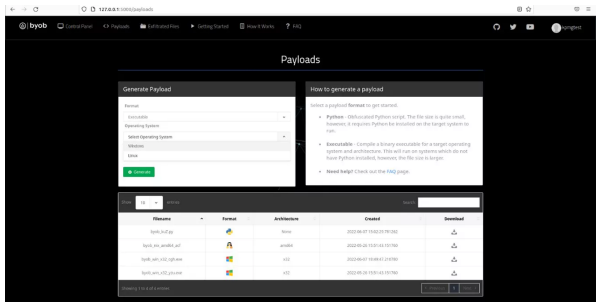
Bring Your Own Botnet (BYOB) is a simple tool which can be setup and executed in just 4 simple steps:

```
git clone https://github.com/malwaredllc/byob.git
chmod +x ./byob/web-gui/startup.sh
./byob/web-gui/startup.sh
```

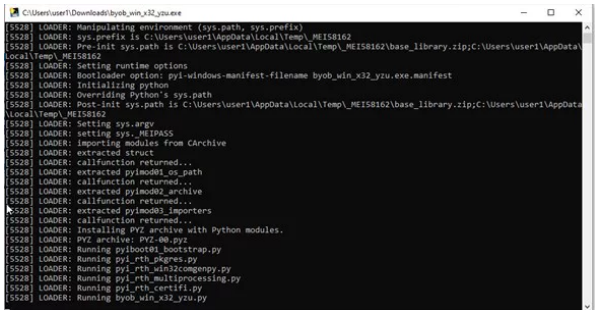
Step 1: Clone the Github repository of BYOB and run the startup.sh script located in the webgui directory.



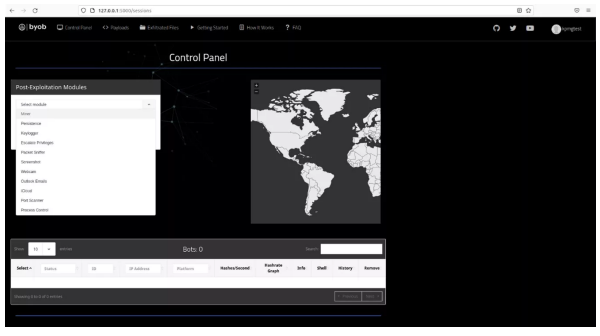




Step 2: Generate a payload for any type of following machines – Windows, or Linux



Step 3: Execute the payload on target machines (in this case



Step 4: Have fun with the Command & Control Center (on GUI).

your own resources).

This GUI will list the bots that called back to the C2 Center. From here you can run different Post-Exploitation Modules to make the bots do what you want.

Exploring various botnet frameworks

BYOB is not the only botnet framework available; there are many similar frameworks that are equally easy to set up. Attackers have started to create their own custom, more sophisticated payloads to run on these readily available C2 servers, often modifying the Python code to alter post-exploitation modules. Some other relevant frameworks include:

- **Covenant:** It is a .NET command and control framework and supports .NET Core – which is multi-platform and due to this Covenant can run natively on Windows, Linux, and MacOS platforms.
- **FactionC2**: Though Faction only supports .NET payloads and modules, but it provides more features than your average C2 framework. Data management is easy with this one, as you can even extract your data through SQL queries!
- **Merlin**: This is the shadiest post exploitation HTTP/2 C2 Server and agent written in Golang since it helps red teamers to evade network detection by using a protocol that the security solutions don't understand or detect. Merlin is also one of the few C2 frameworks which supports Android.

The list can go on but a few honorable mentions have to be iBombshell, FudgeC2, Callidus (uses Outlook, OneNote and MS Teams for command & control), APfell, Sliver and Dali (works on bring-your-own-implant concept with a MySQL backend).

Recommendations

Understanding botnets and building controlled botnets for research purposes contribute significantly to developing effective defense strategies and proactive cybersecurity measures. By exploring the threat landscape, studying attack vectors, and responsibly conducting research, we can better protect ourselves, organizations, and the digital ecosystem from the threats posed by botnets and other cyber-attacks.

By understanding these attack vectors and studying the behavior of different botnets, researchers can enhance their knowledge of cybersecurity threats and develop effective countermeasures. It is crucial, however, to approach the building of a botnet ethically and responsibly. Self-built botnets, intended for research purposes, differ from malicious botnets in terms of IP address diversity. While a malicious botnet comprises compromised devices from various sources, an ethical botnet consists of controlled devices owned by the researcher or organization conducting the research.

If you are facing a botnet attack, need assistance in testing your organization's resiliency against such attacks, or want to strengthen your overall cybersecurity, we are here to help. **Contact our team of cybersecurity experts today for personalized guidance, robust support, and tailored solutions designed to safeguard your systems and data.**

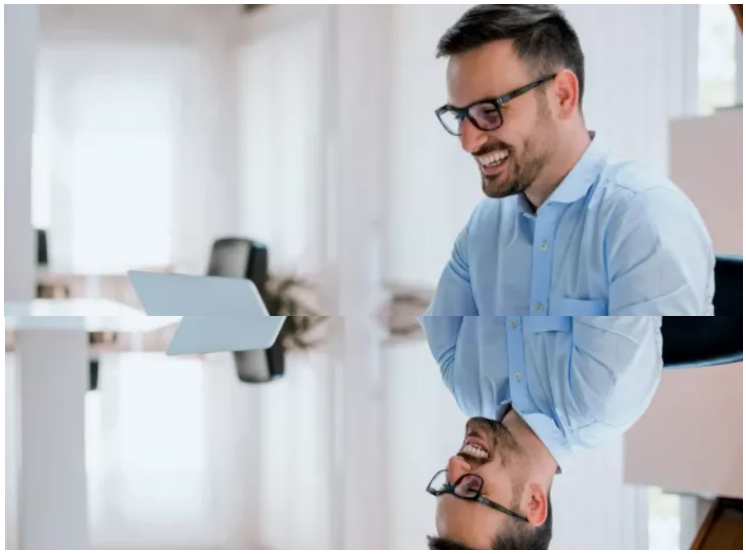
Cybersecurity is an ongoing battle and staying vigilant is crucial. By staying informed, implementing best practices, and working together, we can create a safer and more secure digital environment for everyone. **Take the first step towards strengthening your cybersecurity defenses—reach out to us now!**

When you're interested in more information, please feel free to reach out to us. We're also happy to help you by setting up 'white hat' botnets and assess configured defensive capabilities within your organisation.

Important Notice 1: This article is for educational purposes only. Do not use these techniques against targets without explicit permission of the system owner.

Important Notice 2: This article suggests use of open-source software from public repositories. Please be aware that these projects come without warranty, and disclaiming liability for damages resulting from using the projects.

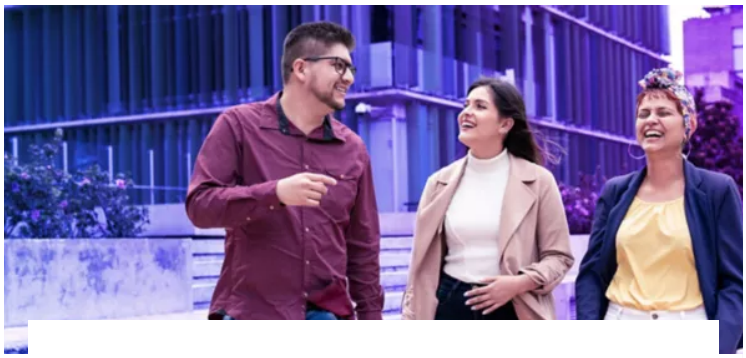
Discover more

Contact us

Frank Wiersma
Senior Tech Consultant, Cyber Offensive Security
KPMG in the Netherlands

Ishan Chandra
Manager, Cyber Offensive Security
KPMG in the Netherlands



Interested in a career at KPMG?

Discover our vacancies

We will keep you informed by email.
Enter your preferences here.

Sign Up ›

Submit RFP