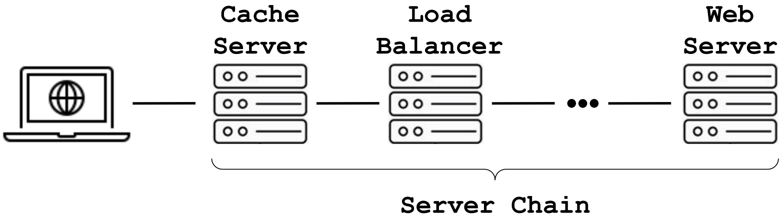
RESEARCH STATEMENT

My research has opened a new frontier in cybersecurity, more specifically in web and network security, by entering an uncharted territory – increased complexity of the content delivery in the modern Web. I have shown how the distributed nature of the content delivery creates an abundance of security challenges. I will continue to discover and counter new security challenges that emerge as the Internet continues to grow and becomes more complex. My research is of interest to **many funding agencies such as National Science Foundation, DARPA, Office of Naval Research, Army Research Lab and Air Force Research Lab, and many Internet companies such as Amazon, Google and Microsoft.**

PRIOR RESEARCH

Exploiting the distributed nature of web content delivery.

On the Internet today, the web content is not delivered directly by the web server. Rather, the content delivery is usually distributed over a chain of servers to enable the functionalities such as caching, security filtering and load balancing (see the figure below). So far, this line of my research has showed three main types of security challenges that are created by this distributed nature of the web content delivery:



1. *Fundamentally new class of cyber attacks*

   Unlike the traditional attacks which exploit a vulnerability on an individual server, server-chain attacks target a chain of servers (see the figure above). In other words, even though each server in the request chain is free of security bugs, the chain as a whole is vulnerable to server-chain attacks due to the conditions that are inherent to chaining servers such as performance optimizations and implementation differences between the servers in the chain. Perhaps, the most destructive and famous of these attacks is HTTP Request Smuggling and it relies on request body parsing differences between servers in a chain. I designed and built a framework for the systematic identification of body parsing differences and demonstrated that a significant portion of those differences lead to Request Smuggling [5]. Another popular mechanism in server chains is the HTTP/2-to-HTTP/1 conversion, which essentially happens when a proxy server (e.g., load balancer) converts incoming HTTP/2 requests into HTTP/1 before forwarding them to the next server (e.g., web server). When we examined the HTTP/2-to-HTTP/1 conversion process on popular servers, we found many conversion anomalies and showed that many of them can be exploited for different types of server-chain attacks such as Request Blackholing and Denial-of-Service (of a specific type which is due to the exhaustion of a connection pool) [4].

2. *Enhanced fingerprinting capabilities for attackers*

   CDN servers usually employ a number of techniques to hide the information of servers running behind them to counter fingerprinting attempts of attackers which usually aim to glean valuable information such as the IP address (e.g., to bypass the security filters on the CDN and directly access the servers) and the server type and version (e.g., to exploit a specific vulnerability). We develop a multilayer fingerprinting technique which takes advantage of request parsing differences between the servers and sends crafted requests to trigger an error in each layer (e.g., first sends a request to trigger an error in the first layer (e.g., CDN server), then sends another request which is forwarded by the first layer, but triggers an error in the second layer (e.g., load balancer), and so on) [9]. The differences in error responses are big enough to enable telling apart the server types (e.g., NGINX) and even the different versions within a server type. This technique improves the state-of-the-art fingerprinting capabilities of attackers substantially.

3. *New approach for circumventing web application firewalls*

   Web application firewalls usually sit in front of the web application servers and inspect every incoming request using a set of string rules matching various attack payloads. Unlike traditional bypass techniques which disguise the attack payload, we exploit the parsing differences between web application firewalls (e.g., Cloudflare WAF) and application frameworks (e.g., Django) and craft requests in a way that the web application firewall cannot parse the attack payload out of the request, whereas the web application framework successfully extracts and executes the attack payload (ongoing research).

This research of mine has been very well received by the software and security community around the world. The developers of Apache Traffic Server (a popular server used by several CDN companies) added the product of my research, the *T-Reqs* tool, to their toolset to proactively search for server-chain attack vectors. The developers of Envoy Proxy had me spend a summer with them at Google to build a toolset for the discovery of the same attack vectors in the Google Cloud. Security teams such as the red team of Hewlett-Packard in Brazil adopted the same tool to test their target systems for these attack vectors. Academic institutions such as CISPA of Germany have included my papers in the curriculum of their web security classes. These papers of mine have also been nominated for the "Top 10 Web Hacking Techniques" award both in 2021 and 2022 by PortSwigger Research [6, 7], and for the "Best Research Paper" award by CSAW at NYU. They were also covered by cybersecurity news media outlets such as *Daily Swig* and *Security Weekly* [2, 1, 8]. Finally, this research has improved the security of the most popular servers on the Internet including in-house servers, such as HAProxy and Apache Tomcat, and CDN servers, such as Akamai and AWS CloudFront, by discovering serious attack vectors and having them eliminated.

Protecting web applications against emerging attacks.

This research of mine aims to improve the defense of web applications against emerging attack vectors. One of the emerging threats is the Server-Side Request Forgery (SSRF) attack which has serious consequences for web applications and even more so for the applications hosted in the cloud systems. In fact, the SSRF attack was the main attack behind the recent CapitalOne hacking incident which resulted in the leakage of 100 million credit card applications and accounts. We introduced a novel defense technique which suggests the deployment of a fetcher service which has no access to the internal services and the rewriting of URLs on the reverse proxy to direct all requests to the fetcher service [3].

Deriving offense and defense techniques from biology.

For the last three years, I have been studying the organ systems in the human body in an effort to learn offensive techniques from the ways diseases arise and defensive techniques from how the body protects itself from various threats. For example, by studying immune cell mechanisms for sensing invaders, I developed a theoretical defense approach against HTTP Request Smuggling attacks. In addition, I developed a mutation strategy in my Frameshifter work by deriving ideas from the frameshift mutations of the DNA in living cells [4].

I also spent about six months at Cold Spring Harbor Laboratory, a biomedical research lab, where I worked with a group of experts from computer science and immunology in order to deepen my experience in biomimetic security research. I studied the most common techniques of human viruses and derived evasion techniques for computer malware from the real viruses. I also studied the immune system mechanism for removing old memory immune cells to make space for the new memory cells, in order to develop a turnover algorithm for IP blacklists in low-memory IoT devices.

RESEARCH VISION

My research agenda is to improve the security of the Internet and distributed network systems by discovering and countering the emerging security challenges. My view is that the growth of sophisticated computer systems will require more systems to become distributed, and the distributed nature will introduce new security challenges like the ones I have found with my prior research. My research vision is to systematically analyze, understand and develop offense and defense techniques in order to secure the emerging attack surfaces.

Expanding research on security challenges of distributed systems on the Web.

The Web has an abundance of systems that operate in a distributed manner. The server-chain security challenges that have been shown by my prior research is the tip of the iceberg. In an effort to expose more of the iceberg, I plan to expand my research in the directions below:

1. *Improving search techniques*

   My main search technique for the server-chain attack vectors has been the blackbox fuzzing. I plan to improve my search process in two primary ways. First, I will take advantage of the source code availability of almost all popular HTTP servers and incorporate various code analysis techniques into the search process, and also improve the efficiency of the fuzzing search (e.g., coverage-guided fuzzing). Second, I will reduce the number of assumptions about the target setup and simulate various real-world server-chain setups, for example by testing various configurations of servers in the chain.

2. *Broadening the attack surface*

   My prior research has mainly looked at the basic components of server chains. However, there are many additional components and factors that determine the behavior of server chains such as HTTP server extensions, web application frameworks, and the architecture of the back-end web application. I plan to broaden my research by asking questions similar to the ones below:

   (a) How do server extensions change the behavior of servers in a chain?

   (b) Does the request parsing of the web application frameworks create new security challenges?

   (c) Does service-mesh architecture of the back-end application introduce new attack vectors?

3. *Moving into server chains of various different protocols*

   HTTP is not the only network protocol where a packet is processed by a chain of servers. DNS protocol with its queries iteratively traversing through multiple DNS servers to resolve a name into an IP address and SMTP protocol with its emails processed by multiple mail servers for security and spam filtering, are two other examples. I plan to investigate the security challenges created by server chains in different network protocols as well.

4. *Measuring various aspects of server chains on the Internet*

   In my recent paper [9], I proposed and helped develop a tool for multilayer server fingerprinting which essentially lets us fingerprint deeper layers of server chains as opposed to the state-of-the-art fingerprinting tools which can only fingerprint the first layer. This tool (and the ones that can be developed for different network protocols) can help us answer various questions about the server chains of various network protocols.

   (a) What are the newest trends in the network topologies employed by websites?

   (b) Do system owners use legacy versions of HTTP servers assuming security by obscurity?

   (c) Can fingerprinting let us reveal the IP addresses of web servers and completely circumvent CDN protections?

Discovering and countering threats to networking in distributed and decentralized systems.

I also plan to use my prior research experience on the distributed systems of the Web and broaden my research to discover and counter the security challenges of networking in distributed and decentralized systems more broadly. These include systems such as robot and drone swarms and sensor networks. I will also leverage my prior experience in bio-inspired security research and seek to learn from the distributed defense solutions of biological systems and develop effective countermeasures against the security challenges of distributed and decentralized systems. I plan to expand this research of mine in the directions below:

1. *Parsing discrepancies in heterogeneous systems*

   Content delivery on the Web relies on the cooperation of servers of different vendors. Since each vendor has its own interpretation of protocol specifications and own implementation choices especially for the parts that are left vague in the specifications, parsing discrepancies between servers are unavoidable and my research has shown that these parsing discrepancies create a set of serious security challenges. Similarly, most of the decentralized systems, such as drone swarm systems and sensor networks, relies on the cooperation of systems by different vendors. I plan to examine the parsing discrepancies in heterogeneous decentralized systems and discover the resulting security challenges.

2. *Network and application level attacks in decentralized systems*

   I also plan to leverage my prior experience in discovering and countering network and application level attacks on the Web and secure decentralized systems against these attacks by asking questions such as those shown below:

   (a) How does the decentralized nature of these systems create unique network level security challenges?

   (b) How can we develop application-level defense mechanisms for systems such as drones and sensors that are not only effective, but also energy-efficient?

   (c) Ground swarm robotics has learned many algorithms from ant colonies. Similarly, can we learn from defense solutions of biological systems for securing decentralized systems such as drone swarms?

References

[1] Jessica Haworth. HTTP request smuggling vulnerability in Apache Tomcat 'has been present since 2015', 2021. `https://portswigger.net/daily-swig/http-request-smuggling-vulnerability-in-apache-tomcat-has-been-present-since-2015`.

[2] Jessica Haworth. New differential fuzzing tool reveals novel HTTP request smuggling techniques, 2021. `https://portswigger.net/daily-swig/new-differential-fuzzing-tool-reveals-novel-http-request-smuggling-techniques`.

[3] Bahruz Jabiyev, Omid Mirzaei, Amin Kharraz, and Engin Kirda. Preventing server-side request forgery attacks. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, pages 1626–1635, 2021.

[4] Bahruz Jabiyev, Steven Sprecher, Anthony Gavazzi, Tommaso Innocenti, Kaan Onarlioglu, and Engin Kirda. {FRAMESHIFTER}: Security implications of {HTTP/2-to-HTTP/1} conversion anomalies. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 1061–1075, 2022.

[5] Bahruz Jabiyev, Steven Sprecher, Kaan Onarlioglu, and Engin Kirda. T-Reqs: HTTP Request Smuggling with Differential Fuzzing. In *ACM Conference on Computer and Communications Security*, 2021.

[6] James Kettle. Top 10 web hacking techniques of 2021 - nominations open, 2021. `https://portswigger.net/research/top-10-web-hacking-techniques-of-2021-nominations-open`.

[7] James Kettle. Top 10 web hacking techniques of 2022 - nominations open, 2022. `https://portswigger.net/research/top-10-web-hacking-techniques-of-2022-nominations-open`.

[8] Mike Shema. Bug Bounties in Windows/WebKit, Edge Hardening, OAuth Hardening, & GoDaddy Breach – ASW #176, 2021. `https://www.scmagazine.com/podcast-segment/bug-bounties-in-windows-webkit-edge-hardening-oauth-hardening-godaddy-breach-asw-176`.

[9] Cem Topcuoglu, Kaan Onarlioglu, Bahruz Jabiyev, and Engin Kirda. Untangle: Multi-layer web server fingerprinting. In *NDSS*, 2023.