Available online www.jsaer.com

Journal of Scientific and Engineering Research, 2023, 10(10):144-147

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

IP Whitelisting: A Critical Server-side Cybersecurity Element

Prashanth Kodurupati

Information Technology, Managed File Transfer Engineer, Minisoft Technologies LLC, Alpharetta, United States of America
Email: prashanth.bachi21@gmail.com

Abstract A whitelist is a list of trusted IPs that are allowed to establish a secure connection with a server. IP whitelisting is the process of limiting server-side connectivity exclusively to the IPs in the whitelist and blocking out every other IP attempt to form a connection. This prevents a wide range of problems, though its primary focus is preventing unauthorized servers from establishing a connection to your server. However, it has its own set of problems, like its limitations when dealing with dynamic IP servers and maintainability. The best case scenario, where IP whitelisting achieves its objective while staying easy to maintain and implement, requires adherence to IP whitelisting best practices.

Keywords *IP Whitelisting*

Introduction

A "secure" internet connection is key to the safe transfer of zettabytes of data every day. So, one of the primary safety measures all servers on the internet employ is ensuring that only legitimate and authorized individuals, servers, or other online entities can establish a connection with them. There are multiple layers to this security approach, and one of them is IP whitelisting. The idea is that the server is inaccessible by default, and no client-server can establish a secure connection with it. A list of IPs considered authorized/legitimate is generated and configured in the firewall (typically with an appropriate script), and only these IPs are given permission to establish a connection with the server. This list of IPs is called a whitelist. IP whitelisting, also called allowlisting or IP filtering, is when a server only allows certain IPs to pass through its firewalls to establish a connection.

Literature Review

References of literature on IP whitelisting go back to the early 2000s. One of the earliest references explored IP whitelisting in the context of email filtering and greylisting, and another identified whitelisting as the earliest technique used to block spam messages. There are also references to it used in internal cybersecurity security, specifically a campus grid [1]. However, the bulk of early literature references on IP whitelisting focus on email spam protection [2]. But later papers, especially in the last four to six years, offered insights on a broader range of use cases for IP whitelisting, like its use in the management of APIs and microservices, which also had limitations, revealing IP whitelisting's inflexibility in certain scenarios [3]. Another interesting reference to IP whitelisting was regarding Unmanned Aerial Vehicle (UAV) security, and the recommendation was that IP whitelisting should be applied at both the device and server level [4]. The literature focused on the defensive use of IP whitelisting also discusses the attack vectors it proves resilient against, like the Address Resolution Protocol (ARP) spoofing.

In the context of IP whitelisting, we can examine two different problem sets:

Problems IP Whitelisting Solves

The overarching problem IP whitelisting solves is preventing unauthorized access to a server, making a connection exclusive to a list of trusted IPs (the whitelist). This makes it a viable protection strategy against a wide range of cyber attacks like brute-force attacks, data breaches, and malware propagation, especially if the whitelist can be updated in real time to remove infected IPs. It also offers limited protection against DDoS attacks.

While it's typically implemented on the server side, the literature suggests use cases involving individual devices as well that have to be isolated from unsafe connection requests.

IP Whitelisting Implementation and Maintenance Problems

There are a few problems associated with configuring IP whitelists and implementation in servers and firewalls. Some prominent problems are:

Dynamic IP Addresses: It's an inherent issue/limitation of IP whitelisting, but it's limited to servers that receive requests from client servers/users who employ dynamic IP addresses that routinely change. If the IP address of a client-server changes, the IP whitelist of your server has to be updated to reflect this change. If the frequency of this change is too high, it may be a major maintainability concern [5].

Whitelist Rules: Whitelist rules can be straightforward to implement when the goal is simply to exclude all IPs other than the ones included in the whitelist. However, it can be a challenge when the rules restrict or permit data transfer, retry attempts, or have to govern automated list updation via third-party tools [6].

Maintainability: Maintaining an IP whitelist, especially when client servers frequently change their IP or rules need to be changed to accommodate different clients with their own restrictions and security elements, can become a challenge.

UAT and Production Environment Issues: While implementing an IP whitelist, it's important to ensure that the lists and rules mirror each other in UAT and production environments. Otherwise, you may experience unforeseen challenges. Similarly, while whitelisting can prevent unauthorized connections, it alone may not be enough as a safeguard against malicious data flowing in from trusted sources. These scenarios should be addressed in both environments.

Solution: IP Whitelisting Best Practices

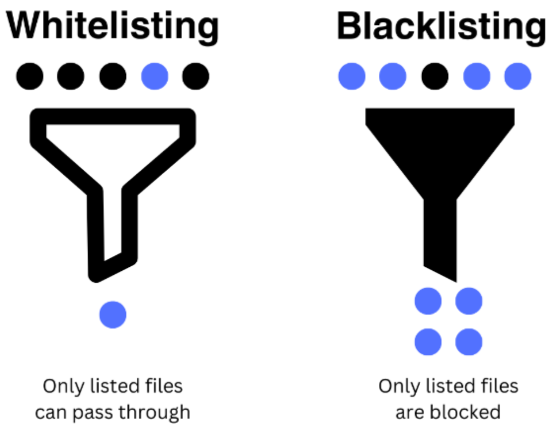
The solution to both sets of problems is to adhere to the best practices of IP whitelisting. When properly designed and implemented, an IP whitelist doesn't just protect against unauthorized access from all other IPs; it may add a few other layers of protection as well. However, that's reliant upon the sophistication of the firewall itself and the IP whitelisting rules.

Understanding and Navigating IP Whitelisting Limitations

One of the first and most important IP whitelisting best practices is to understand its limitations, use cases, and its place in the overall server security framework/infrastructure. If your server needs to entertain requests from a wide range of client servers that use dynamic IPs, then a basic IP whitelisting script where the list is manually maintained is not feasible [7]. However, if this list is automatically updated with new IP addresses through trusted intermediaries or dynamic IP client servers are handled differently from static IP clients, then IP whitelisting can be a powerful security strategy.

Using IP Whitelisting in Conjunction with Blacklisting

Another good strategy is using IP whitelisting and blacklisting strategies together, creating an additional layer of security. A whitelist serves as a filter that only allows certain IPs to establish a connection and blocks everything else. In contrast, a blacklist only stops certain IPs while letting every other IP through [8]. Hence, the typical stacking is IP blacklists preceding the whitelists so that known malicious entities are filtered even before reaching the whitelists. The blacklists can be maintained and updated through collaborations within specific networks (like banks, financial institutions, etc.) or cybersecurity vendors and can be more comprehensive in nature. Meanwhile, whitelists can be maintained internally and updated.


Whitelisting
Blacklisting
Only listed files can pass through
Only listed files are blocked

Maintainability-Oriented Whitelist Design and Implementation

When developing and implementing an IP whitelist, make sure long-term maintainability is factored into the development phase. If you build a security strategy to prevent unauthorized access and the script is not flexible enough to accommodate a growing range of clients or more sophisticated rules, it may become time-consuming and difficult to maintain over time. Similarly, the implementation of an IP whitelist has to take into account a comprehensive range of use cases. When a whitelist is implemented on just your server, it may be easy enough to maintain, but if you are serving as an intermediary, the implementation characteristics will differ. If your server has a whitelist and a company in your whitelist has to connect with another company (with your server acting as a pass-through entity) to access or transfer data, you may have to request a reconfiguration of the host server's whitelist (in some cases).

Simplify Whitelisting Rules

The more complex rules you set when developing and implementing your whitelist, the more difficult it may be to maintain over time and the more issues it may generate in both UAT and production environments. For example, you don't need to add geo-IP filtering or time-based IP filtering unless you absolutely need to since it may block legitimate connection requests in certain conditions that you may not have communicated to the client in advance, which may lead to operational friction.

However, simplification doesn't mean you can't take full advantage of the flexibility and security the rules can offer. Whitelists can also deny client servers access to specific applications/application servers, and the rules can enforce other layers of restriction. For example, a whitelist might only allow a client-server to establish a connection if it follows the Secure File Transfer Protocol (SFTP). This may prevent any unsafe file/data transfer, even from a whitelisted IP.

Take Advantage of IP Ranges and Subnets

If your client-server uses dynamic IPs or has multiple servers and, in either case, a predefined set of IPs, you can whitelist the entire range to avoid reconfiguring the whitelisting script or updating the whitelist several times. You can also allow for subnets so that different users, servers, or other online entities that are part of a trusted network are not blocked for having the right IP address. Subnets can also allow you more granular control over who can access your server or establish a secure connection with it from your client side.

Results and Potential Use Cases

The desired result of a well-developed, properly implemented IP whitelist is that it will prevent any unauthorized user or server from passing through your server’s firewall. In addition to making your servers more secure, it can also reduce the attack surface, reducing the overall burden of your cybersecurity.

As for potential use cases:

Prerequisites	Example Entities
Servers/Organizations with Static IP Clients	Financial institutions
Limited Set of Authorized Connections	Internal servers
High-Security Environments	Research facilities Healthcare servers (patient data)
Known and Trusted Partners	Research collaborations
Clearly Defined Access Needs	Any organization with strict access control

Conclusion

IP Whitelisting is a critical line of defense against unauthorized accesses, especially in static IP environments and several security-critical use cases. Its implementation comes with its own set of challenges but they can be averted/mitigated by following IP whitelisting best practices. Its potency as a cybersecurity element can be enhanced by combining it with complementary defenses and leveraging features like IP ranges and subnets.

References

[1]. Derek Weitzel, Brian Bockelman, Dan Fraser, Ruth Pordes, David Swanson; Enabling Campus Grids with Open Science Grid Technology; *Journal of Physics: Conference Series, Volume 331, Issue 6* (2011); https://iopscience.iop.org/article/10.1088/1742-6596/331/6/062025/meta

[2]. Zahra S. Torabi, Mohammad H. Nadimi-Shahraki, Akbar Nabiollahi; Efficient Support Vector Machines for Spam Detection: A Survey; *(IJCSIS) International Journal of Computer Science and Information Security, Vol. 13, No. 1* (2015); https://www.academia.edu/download/37127910/03_Paper_31121412_IJCSIS_Camera_Ready_pp._11-28.pdf

[3]. Oras Baker, Quy Nguyen; A Novel Approach to Secure Microservice Architecture from OWASP vulnerabilities; *Proceedings of the 10th Annual CITRENZ Conference* (2019); https://www.researchgate.net/profile/Emre-Erturk-3/publication/337367691_2019-CITRENZ-Conference_Book/links/5dd46deba6fdcc37897a4eb9/2019-CITRENZ-Conference-Book.pdf#page=56

[4]. Susheela Dahiya, Manik Garg; Unmanned Aerial Vehicles: Vulnerability to Cyber Attacks; International *Conference on Unmanned Aerial System in Geomatics, Proceedings of UASG 2019 pp 201–211* (2019); https://link.springer.com/chapter/10.1007/978-3-030-37393-1_18

[5]. NordLayer; What is IP allowlisting (whitelisting)? Understanding the basics and beyond; *NordLayer Blog* (2021); https://nordlayer.com/blog/ip-whitelisting-for-cloud-security/

[6]. Rongfeng Zheng, Jiayong Liu, Weina Niu, Liang Liu, Kai Li, and Shan Liao; Preprocessing Method for Encrypted Traffic Based on Semisupervised Clustering; *Machine Learning and Applied Cryptography* (2020); https://www.hindawi.com/journals/scn/2020/8824659/

[7]. Ankit Kumar Jain, B. B. Gupta; A novel approach to protect against phishing attacks at client side using auto-updated whitelist; *EURASIP Journal on Information Security* (2016); https://link.springer.com/article/10.1186/s13635-016-0034-3

[8]. Ibrahim Ghafir, Vaclav Prenosi; Blacklist-based malicious IP traffic detection; 2*015 Global Conference on Communication Technologies (GCCT), IEEE* (2015); https://ieeexplore.ieee.org/abstract/document/7342657