Misconfiguration in Firewalls and Network Access Controls

Subjects: Computer Science, Information Systems
Contributor: Michael Alicea

Firewalls and network access controls play important roles in security control and protection. Those firewalls may create an incorrect sense or state of protection if they are improperly configured. One of the major configuration problems in firewalls is related to misconfiguration in the access control rules added to the firewall that will control network traffic. The three most common misconfigurations are shadowing, correlation and redundancy.

Keywords: firewall roles misconfiguration

1. Shadowing

In a firewall rule list, Reference [1] states that “A rule is shadowed when a previous rule matches all the packets that match this rule, such that the shadowed rule will never be activated”. A more generic rule with a wide scope is preventing the activation of a second rule with a more narrow scope. This definition, or one very similar to it, is present in every paper discussing the topic of rule shadowing. This definition is fairly generic, and we can see that some authors add one key condition. The author [2] gives the additional condition that the two rules must have different actions, if the shadowed rule is an allow, the shadowing rule must be a deny and vice versa. This distinction is important as it is what differentiates shadowing from other misconfigurations. With this definition, we can now observe exactly how this misconfiguration occurs. Traditionally in firewalls, rules are enforced in the order they occur in the list of rules. Shadowing will occur when a rule higher in the list completely overlaps a rule lower in the list. This makes rule ordering something to take into careful consideration [3]. When you have a rule with a scope that is entirely contained in another rule, shadowing may occur depending on which order they occur in. Having the correct rules makes no difference if they are in the wrong order. Some firewalls have an option to add a numbered priority onto rules to set the order in which they are enforced. An example of a firewall with this implementation would be the firewall in Google Cloud Platform. Instead of relying on the order in which the rules are written, the rule with a higher priority can potentially shadow a rule with lesser priority. In a firewall that enforces rules on a numbered priority system, rule order is completely irrelevant. In this case, take care to assign the correct priorities and know how the firewall handles rules with the same priority.

An IDS/IPS such as Snort can suffer from these issues as well, but the reason they occur is different. Reference [4] lays out that rule priority inside of Snort is determined by the action associated with the rule such as log, alert, pass, etc. Shadowing occurs in Snort when one rule has a scope that encompasses a second rule and has a higher-priority action. While action taken has no impact on the order of rule enforcement in a firewall, an action in Snort can only shadow an action of lower priority. For example, an alert rule in Snort cannot shadow a pass rule, but a pass rule can shadow an alert rule. Due to the higher priority of the pass action, a pass rule will always be evaluated before an alert rule. In the event that a packet matches two or more rules with the same action, every matching rule will be triggered. For instance, if an incoming packet meets the criteria for multiple log rules, each of those rules’ alerts will be logged. This trend will continue with the other misconfigurations listed in this SLR, Snort suffers from the same potential misconfigurations, but they occur for slightly different reasons.

Shadowing does not only occur when one rule completely encompasses another rule. The authors of [5] list a different kind of shadowing that they call total shadowing. The authors describe total shadowing as when the combined scope of multiple rules covers the entire scope of another rule. As an example, suppose we have a simple port blocking rule that blocks inbound traffic between ports 1 to 100. A shadowing set could be one allow rule from 1 to 50 and a second from 51 to 100. This causes the initial block rule to be shadowed by the other two rules combined. As we will see later in this SLR, total shadowing actually refers to multiple instances of another misconfiguration, but the end result is the same, and a rule is being prevented from being evaluated.

So far, the discussion around the concept of shadowing has revolved around the assumption that there is only one network access control mechanism present on the network. Larger networks can contain multiple firewalls or IDS/IPS that could conflict with one another. Reference [6] discusses the possibility of rules within two different firewalls on the same network conflicting with one another. When a rule in one firewall shadows a rule in a second firewall, we have what is called an inter-firewall shadowing anomaly. Individually, the two firewalls in question can have perfectly valid rule sets with no misconfigurations individually. The issue is when the rule sets put together cause shadowing. This will result in parts of the network suffering from the results of the misconfiguration. For example, if a firewall at the network edge blocks social media but a firewall for a specific network allows it, only the users behind the second firewall will suffer the effects of the misconfiguration. It is important to note that such an anomaly does not necessarily have to occur between two of the same type of device, i.e., two firewalls or two IDS. What this tells us is that you cannot look at every component individually. Having a well-made network access control policy on individual devices does not necessarily mean the overall network access control policy works as intended. Network access control policy needs to take place on a macro level. Whether or not the sum of all policies mesh together well is just as important as the individual components in the network.

Shadowing is a serious misconfiguration as its existence in a rule set completely nullifies other rules. While this can occur intentionally, such as leaving old rules, unintentional shadowing can lead to many problems. This could cause a denial in legitimate traffic, which could hurt an organization's productivity and cause issues for users. Even worse, this could cause malicious traffic to get through since the rule that would block them is shadowed [7]. Since a shadowed rule cannot be triggered, its removal has no effect on a firewall rule set. The issue is that simply removing a shadowed rule may not be the best course of action if it is supposed to catch some traffic. Alternatively, it may be more desirable to unshadow a rule by limiting the scope of the shadowing rule. When shadowing is detected within a rule set, it is important that the issue is identified as quickly as possible because an accidental inclusion of shadowing can be devastating [8]. Whether the rules are denying good traffic or allowing bad traffic, this kind of anomaly is at best annoying to users and at its worst a threat to the entire network. The high severity of shadowing stems from the fact that it causes the wrong action to be taken even though there is no problem with any individual rule that would throw an error by the firewall or IDS/IPS. Shadowing and total shadowing can be difficult to diagnose without the use of automated tools for this reason, and therefore it is important to evaluate the impact of the interactions of old rules with new rules.

2. Correlation

A correlation anomaly is similar to shadowing. Reference [9] defines correlation as when "one rule matches some of the packets that another rule may capture, and that other rule matches some packets, which the original rule captures, the rules may be in correlation. However, the actions of the two rules need to be different". In short, correlation can be thought of as partial shadowing. Two rules have a partially overlapping scope, causing the higher-priority rule to catch traffic that meets the criteria in that overlap. Only in the overlapping part of the two rule's respective scopes will there be an issue. This is what separates correlation from shadowing, i.e., the fact that a rule is unintentionally overridden some of the time rather than all of the time.

How this anomaly occurs is also very similar to that of shadowing. An earlier rule in a rule list is enforced first before the correlating rule later in the list. As [10] notes, switching the order of the two rules inverses which rule overrides the other in their shared scope. Regardless of which rule overrides the other, the fundamental cause of correlation is that two rules have a partially overlapping scope. The easiest way to think of this is as a Venn diagram, the portion where the two circles overlap is where the issue occurs. Assuming that this overlap is not deliberate, this means that the wrong action is being taken in this shared scope.

Earlier, we discussed how the concept [5] describes as total shadowing. We mentioned before that total shadowing represents multiple instances of a different misconfiguration. Total shadowing is the result of multiple instances of correlation that together entirely cover the scope of another rule. This results in that rule being effectively shadowed as the other rules will not permit this rule to be evaluated when a packet that meets its criteria comes through. Just like normal shadowing, removing this shadowed rule causes no change in the overall policy. Again, this may not be the desired solution if the shadowed rule is wanted. Whether the correlating rules are causing traffic to be denied or allowed when that action should not be the case, these rules should be carefully evaluated to make the correct change to achieve the desired result.

Just like shadowing, a network access policy can suffer from inter-firewall correlation. Reference [11] discusses how rules in multiple firewalls can correlate with one another and cause issues for portions of a network where the rules clash with one another. When multiple firewalls are involved, an upstream firewall can block or allow traffic in which a downstream

firewall takes a different action but the two rules on each firewall do not completely overlap one another. This is equally true when using different types of components in the network, where an IPS may block or allow traffic it is not supposed to going to a downstream firewall. Different subnets inside a larger network will have their own policies in regards to what traffic is allowed in and out, and this may contradict the policy of upstream firewalls resulting in traffic not making it to the downstream firewalls in the first place. Fixing this issue will involve making sure that there are rules in place on upstream firewalls that are specific to certain subnets to avoid any such correlation between the firewalls.

Correlation is a serious misconfiguration for the same reasons as shadowing: it causes the wrong action to occur on traffic. In the policy visualization tool called PolicyVis, Reference [12] describes the issue of correlation as occuring when two rectangles when placed in 2D space have some, but not total, overlap. In a visualization tool like theirs, this is quite easy to notice when two such rectangles overlap. Without tools, however, correlation can be difficult to diagnose. In the event of shadowing, it is very clear that something is being allowed or denied every time when it is not supposed to. Correlation creates a situation where the policy will appear to sometimes work. This inconsistent appearance can be problematic due to the fact that the rule may work properly for some people or machines and not others. This could lead to the assumption that the problem lies with a specific computer's configuration or some users' actions. Once it is discovered that the problem lies within firewall rules, changes will need to be made so that the overlap is either removed, a rule is made more specific to not block the intended traffic or the rule order is reversed so that the correct action is being taken.

3. Redundancy

Redundancy is the least severe of the misconfigurations. Redundancy is as simple as it sounds: it is when two different rules cover the same potential packets and take the same action [13]. Similar to shadowing, this can take place as two identical rules or as a general rule followed by a more specific one. Say, for instance, a firewall has two rules that block ports 20–30; these rules are redundant because both rules perform the same actions on the same packets. It is also a case of redundancy if a rule blocks that same range of 20 to 30 but a second rule only blocks port 22. Since port 22 is in the range of 20 to 30, there is no need to have another rule blocking port 22 since it is already blocked by the first rule.

Redundancy is not as straightforward when it comes to inter-firewall or inter-component misconfigurations. Reference [6] identifies an inter-firewall redundancy anomaly when a downstream firewall blocks traffic that an upstream firewall is already blocking. A rule-blocking specific traffic from outside the network is not needed if an upstream firewall already does so. If there is concern for insider threats, such traffic could occur if the attacker is on the internal network already making the need for some redundant rules between multiple firewalls necessary. This will still cause that rule to be evaluated twice on outside traffic but could nonetheless protect from traffic from an attacker within the same network. Whether or not inter-firewall redundancy is a misconfiguration depends entirely upon the threat level from within the user's own network.

Redundancy is one of the few misconfigurations that has no impact on the overall security of a network. Having multiple copies of the same rule, even across multiple devices does not affect policy enforcement. The only effect redundancy has on a network is to performance [14]. The issue with redundant rules is that it slows down enforcement. A firewall will try to match a packet to one of its existing rules and if it does not match any, it takes a default action. The problem with redundancy is that whenever a packet comes in, it gets checked by more rules than it should have to. Every time a packet is checked against a redundant rule, time and performance of the device is wasted. Redundant rules can be safely removed from a firewall as their removal will have no impact on security and will result in better performance from the firewall.

4. Irrelevance

Like redundant rules, irrelevant rules are of little to no severity to the security of a network. A rule is irrelevant if it is impossible for the firewall to encounter a packet that matches it [7]. Examples of an irrelevant rule could be blocking packets from a subnet that has no route to this particular firewall. If communication is outright impossible due to lack of a physical/wireless connection or otherwise, there is no need to make any rules regarding that subnet. Another example could be blocking traffic from the internet when hosts outside the user's network cannot reach their subnet. If no hosts behind a particular firewall have port forwarding or some way to be accessible from the internet, there would be no need to block traffic from the internet or from certain IP addresses out on the internet because outside hosts cannot send packets to the hosts behind the firewall.

Unlike redundancy, irrelevance cannot occur because of the configuration of other devices. If an upstream firewall blocks traffic that a downstream firewall also blocks, that situation is not irrelevance. That traffic cannot reach the downstream firewall, but it is considered to be inter-firewall redundancy because the traffic is blocked in two different firewalls. The distinction is that irrelevant rules are irrelevant because even in the absence of any rules, the traffic is still impossible [15].

Irrelevant rules are another one of the misconfigurations with zero impact on a network's security. Whether or not a network has irrelevant rules will not impact the security of the network. In addition, like redundant rules, the largest impact of irrelevant rules is on the performance of the firewall [16]. Irrelevant rules suffer from the same issues where their very existence slows down the firewall with no benefit to the security of the network. If a rule is found to be irrelevant, it can be safely removed without any effect on the overall security on the network, and doing so can only be beneficial to network speeds.

5. Generalization

Rule generalization is a misconfiguration that is essentially the opposite of shadowing. Generalization is defined by [17] as when a rule with a more narrow scope has higher priority than a second rule with a scope that encompasses the first. If the rules were reversed, you would have shadowing. An important distinction in generalization is that the second rule must have a larger scope than the first. Using a similar example to that from redundancy, the first rule allows port 22, while the second rule blocks 20 to 30. The different action means that instead of redundancy, the second rule is a generalization of the first.

Generalization is quite similar to correlation. The authors of [17] define generalization as "Suppose the first rule matches the packets that also match the second rule while performing different actions, the rules are then considered generalized". Earlier we gave the example of a Venn diagram to describe correlation. Generalization is like having a smaller circle inside a bigger circle. It can be thought of as "poking holes" inside of another rule.

The impact that generalization has is similar to that of correlation. One rule is partially covering another rule. This can cause issues depending on what the the covered rule was blocking or allowing. Just like correlation, this could potentially lead to some good traffic getting blocked or some bad traffic being allowed. This misconfiguration is, however, the most likely to be intentional. As [18] puts it, "Generalization is considered only an anomaly warning because inserting a specific rule makes an exception of the general rule, and thus confirming this action by the administrator is important". In their tool that detects and corrects firewall misconfigurations, they do not automate fixes for generalization. Oftentimes, network administrators will create a broad rule then have exceptions for certain allowed traffic. In fact, an implicit deny all is a general rule to every single allow rule; thus any rule inside of a firewall that allows traffic is generalized by the implicit deny all. For these reasons, generalization is not always harmful to access control policy so long as its presence is deliberate.

6. FlowBit Misconfiguration

This final misconfiguration has the most variety in exactly what it is and how it occurs. In advanced firewalls and in IDS/IPS such as Snort, there is a feature that allows the user to create rules based on the state of the application protocol. One such example of this is Snort's flowbit feature, which is capable of keeping track of an application state. A flowbit can keep track of whether a user is logged in, what they are doing inside an application, etc. The exact implementation is different, depending on the platform and even the version of the platform. While different devices may refer to this feature by different names, we will refer to this feature as "flowbits", Snort's implementation.

In Snort, flowbits replaced the now-depreciated activate and dynamic rules. Flowbits allow Snort to keep track of what has happened in a session. Rules checking flowbits will only fire if the associated flowbit is set on that session. As [4] explains, there are multiple ways a flowbit rule can be misconfigured. A flowbit could be set to check for a connection state that is impossible, such as being logged in and not being logged at the same time. There is also the issue of not checking for enough connection states. It is possible that any given attack can be carried out in different ways that would have different connection states. While no single rule would be misconfigured, the policy as a whole is misconfigured as it did not take into account other means of that same attack occurring that one or more of its rules are attempting to block. There is also the issue of checking for flowbits that are too specific or broad. Checking for overly broad conditions, say being logged in, may result in many false positives as normal traffic triggers the alerts. Alternatively, too specific conditions could allow an attack to sneak by because they did not meet everything that the flowbits were flagging.

Checking for a connection state can be fairly tricky and thus could be easy to misconfigure. The nature of flowbits often means that rules that check flowbits or their counterparts in other systems will often be looking for a few specific attacks.

The result of a flowbit rule being misconfigured will leave a network either vulnerable or label good traffic as that type of attack. As stated above, the exact impact is determined by how the misconfiguration occurs. It can result in the attack slipping by entirely or causing benign traffic to trigger the rule.

References

1. Chandre, P.R.; Surve, R.R.; Badhan, S.R.; Surve, A.B.; Mane, V.T. Anomalies of Firewall Policy Detection and Resolution. Int. J. Eng. Res. Appl. 2014, I, 696–701.
2. Clark, P.G. Firewall Policy Diagram: Novel Data Structures and Algorithms for Modeling, Analysis, and Comprehension of Network Firewalls. Ph.D. Thesis, University of Kansas, Lawrence, KS, USA, 2013.
3. Al-Shaer, E.S.; Hamed, H.H. Modeling and management of firewall policies. IEEE Trans. Netw. Serv. Manag. 2004, 1, 2–10.
4. Tran, T. Misconfiguration Analysis of Network Access Control Policies. Master's Thesis, University of Waterloo, Waterloo, ON, Canada, 2009.
5. Aryan, R.; Yazidi, A.; Engelstad, P.E. An incremental approach for swift openflow anomaly detection. In Proceedings of the 2018 IEEE 43rd Conference on Local Computer Networks (LCN), Chicago, IL, USA, 1–4 October 2018; pp. 502–510.
6. Taibah, M.; Al-Shaer, E.; Hamed, H. Dynamic Response in Distributed Firewall Systems. Technical Report, DePaul CTI Technical Report, CTI-TR-05-002. 2004. Available online: http://facweb.cs.depaul.edu/research/TechReports/TR05-002.pdf (accessed on 15 October 2021).
7. Zhang, Y.; Zhang, Y.; Wang, W. Optimization of Firewall Filtering Rules by a Thorough Rewriting. In Proceedings of the 4th Latin American Network Operations and Management Symposium, Porto Alegre, Brazil, 29–31 August 2005; pp. 77–88.
8. Al-Shaer, E.; Hamed, H. Design and Implementation of Firewall Policy Advisor Tools. DePaul University, CTI. Tech. Rep. 2002. Available online: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.134.3344&rep=rep1&type=pdf (accessed on 15 October 2021).
9. Geeringh, C. Generic Firewall Rule Compiler And Modeller. Master's Thesis, Napier University, Edinburgh, UK, 2007.
10. Chao, C.S.; Liu, A.C. An internet firewall policy verification system. In Proceedings of the 9th Asia-Pacific Network Operations and Management Symposium, Busan, Korea, 27–29 September 2006; Volume 1, pp. 364–374.
11. Al-Shaer, E.; Hamed, H.; Boutaba, R.; Hasan, M. Conflict classification and analysis of distributed firewall policies. IEEE J. Sel. Areas Commun. 2005, 23, 2069–2084.
12. Tran, T.; Al-Shaer, E.S.; Boutaba, R. PolicyVis: Firewall Security Policy Visualization and Inspection. In Proceedings of the 21st conference on Large Installation System Administration Conference, Dallas, TX, USA, 11–16 November 2007; Volume 7, pp. 1–16.
13. Chao, C.S.; Yang, S.J. A Novel Mechanism for Anomaly Removal of Firewall Filtering Rules. J. Internet Technol. 2020, 21, 949–957.
14. Hu, H. Assurance Management Framework for Access Control Systems. Ph.D Thesis, Arizona State University, Tempe, AZ, USA, July 2012.
15. Ahmed, Z.; S Askari, S.M. Firewall Rule Anomaly Detection: A Survey. Int. J. Comput. Intell. IoT 2018, 2, 6 pages.
16. Thwin, L.W.; Aye, Z.M. Classification and Discovery on Intra-Firewall Policy Anomalies. Natl. J. Parallel. Soft Comput. 2019, 1, 235–242.
17. Vanikalyani, G.; Avinash, P.; Pandarinath, P. Cross-domain search for policy anomalies in firewall. Int. J. Comput. Appl. 2014, 104, 20–24.
18. Al-Shaer, E.S.; Hamed, H.H. Firewall Policy Advisor for Anomaly Detection, Rules Editing and Translation. Int. Symp. Integr. Netw. Manag. 2003, 118, 17–30.