Running head: Network Security

Title of APA Paper

Zaoxing Liu Chang Liu

Department of Computer Science

Johns Hopkins University

Abstract

According to section 1.07 of the *APA Publication Manual* (2001), “An abstract is a brief,

comprehensive summary of the contents of the article; it allows readers to survey the

contents of an article quickly, and like a title, it enables abstracting and information

services to index and retrieve articles” (p. 12).

.

Title of APA Paper

**Give a brief, high-level description of your implementation.**

In this project we developed a front-end web application which is scripted in Ruby on Rails. Our implementation is based on Snort so that this integrated tool could monitor and analyze the flow of the network as well as the security of the network. With Snort as sensor, we could define our own rules to identify the risky activities such as illegal login attempt to the server or Denial of Service attack and then display and notify those risky activities to the user in our front-end.

Our tool has the ability to perform real-time traffic analysis and packet logging on Internet Protocol (IP) networks. At the same time, the tool can produce some analysis charts and graphics to present the network traffic and network attacks, such as SYN Floods, DNS Spoofing attacks and buffer overflow in the system which will make it easy for the system administrator to figure out the potential security vulnerabilities in their system. For the demo of our application, please refer to the Appendix A.

**Indicate the final results**

We tested our implementation by performing several test cases, all the test cases had been successfully performed and passed, this means our design and implementation is correct. The table below showed the attack tests we had made and the result of these test. We obtain the normal network traffic by using the system normally, thus no specific test is performed.

For details of how we perform a test case and how our application handle such a test, please refer to the Appendix A

|  |  |  |
| --- | --- | --- |
| Test Case | Output | Result |
| Invalid Login Attempt | The system successfully filtered out this kind of attempt and displayed it in the system. | PASS |
| Denial of Service Attack | The system successfully detected, blocked and displayed it in the system. | PASS |
| ARP Attack | The system successfully detected, blocked and displayed it in the system. | PASS |
| Policy Violation | The system successfully detected and displayed it in the system | PASS |
| Unauthorized Root Access | The system successfully detected and displayed it in the system | PASS |

**Final conclusion**

Our system is a functional and easy application that makes simple network security monitoring possible, it could recognize and analyze unexpected access to the network by modifying the snort platform. Our own front-end application could produce analysis charts and graphics to both current and historical network traffic. A network or system administrator could easily gain all the information on the network by using the user-friendly graphic interface. Our system could also alert any usual network activities to help the administrator find the potential vulnerability in the current system. Also, the modified rule matching algorithm makes our new system performs better than the old Snort platform.

Appendix

1. **A brief application demo**

In this section, we will demo the basic function of our web application.

On the home page, our application will show a chart of the total number of security alerts during the past day, as showed in the figure A.1.

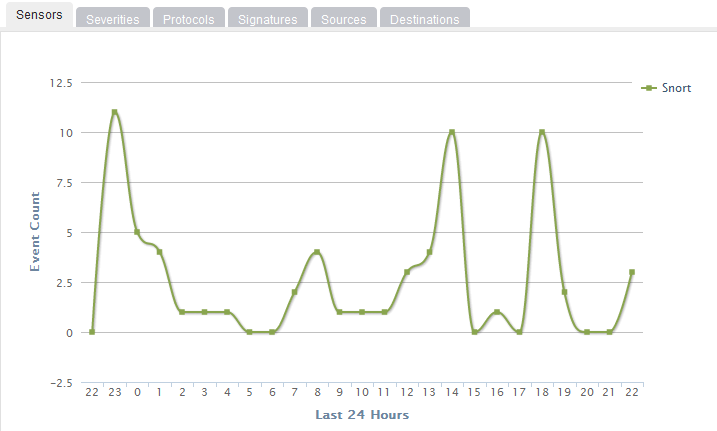


Figure A.1 Chart of total number of security alert

We can shift to each tab to get the detailed summary in different fields. For example, the Protocol tab will show all kinds of the protocols that had been used during the 24 hours, as showed in the figure A.2.

In the Events page, we could see the detailed events information instead of the summary on the front page. Figure A.3 shows how the detailed events list looks like. The orange tag with number 2 means this activity may not necessary be a secure or a risky event, but administrator should pay attention to it. Other kind of tag includes red tag with number 1 and green tag with number 3, a red tag means the event may be an attacking attempt while green tag means such an event is a normal and safe event. With the different color we defined in our application, a system administrator could quickly tell if an activity is risky.

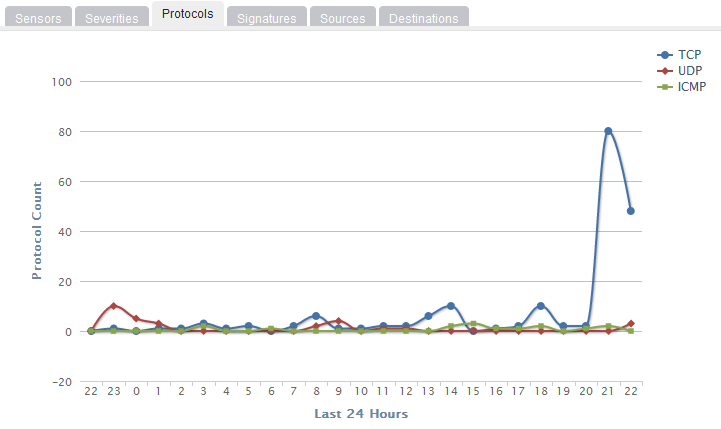


Figure A.2 Chart of different protocols

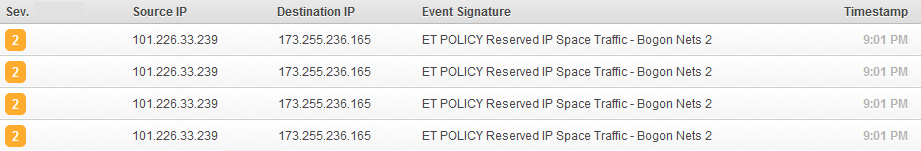


Figure A.3 Detailed events list

We can obtain further information of each record by click on the record, a new page with details of that record will show up, as in the figure A.4. The detailed information includes IP Header information, Signature information, TCP header information, Payload, etc. These information will be helpful when the administrator want know more about a certain records.

We would also like to include an example of how our system will record a simple attack attempt in our demo. If a remote user trying to login to the server by guessing the password of a certain user, this should be considered as the password guessing attack. Our application will identify this kind of attack and assign it with the red tag, which means this kind of login attempt may be an attack to the system and the system administrator should pay attention to it. A typical record of this kind is showed as figure A5.

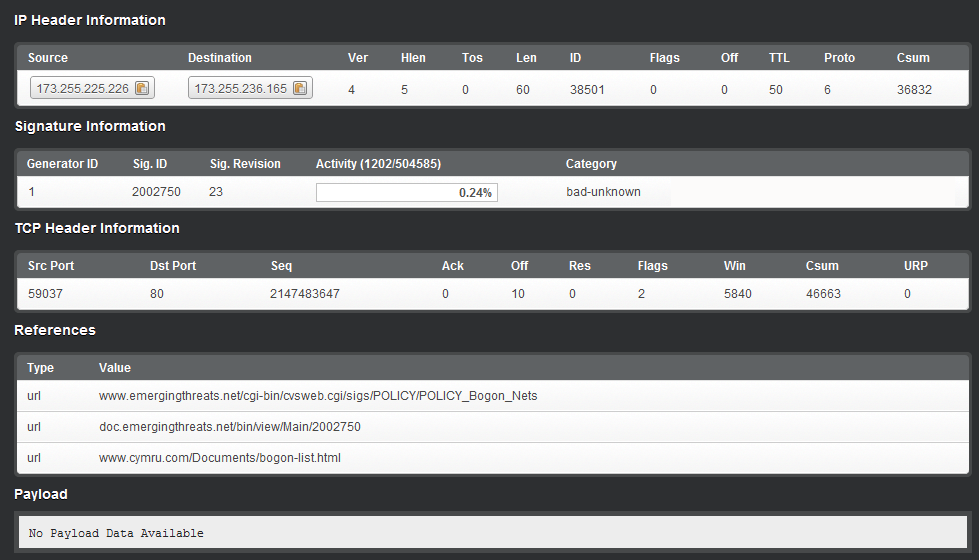


Figure A.4 Detail information of an event



Figure A.5 Records of bad login attempts





References

//TODO