Scripting an Automated Score and Message Board;
Cybersecurity Competitive Labs as a Service (CLAaaS)
Nicholas Hempenius and Te-Shun Chou

Background
This research discusses an Automated Score and Message Board (ASMB) system used in a Competitive Labs-as-a-Service (CLAaaS) education platform. The CLAaaS platform provides students with identical simulated networks containing multiple Virtual Machine (VM) servers. The student networks are interconnected in a larger self-contained virtualized environment. In CLAaaS, students harden their own VM servers and attack others. The ASMB displays and updates student scores when servers are successfully hardened or attacked. CLAaaS encourages a high level of interaction between students and the ASMB facilitates competitive learning among students.

Introduction
This presentation describes the methods and technologies behind the ASMB used in the CLAaaS system. The inherent nature of cybersecurity encourages bending rules. This meant security needed to be considered when designing the ASMB. The ASMB was designed as a five-step process.

1. **Collection**: Where and how to securely collect scoring metrics?
2. **Transportation**: Where and how to securely transport the metrics?
3. **Storage**: Where and how to securely store the metrics?
4. **Analysis**: Where and how to securely analyze the metrics?
5. **Display**: Where and how to display the results?

Collection
The purpose of this step was to create a flow of data to be analyzed. Custom scoring agents were created to run on each VM server. The agents run checks in periodic intervals, collecting information from data points determined by lab objectives and monitor for objective completion. The scripts run only in memory as Terminate Stay Resident (TSR) programs and are hidden from even privileged users. Figure 1. below is an example of such code that might appear in the agent scripts.

```
1: Check-Agents-Objective1:
2: IF FIREWALL = (Check if the firewall is on)
3: THEN
4: IF FIREWALL = ON
5: THEN
6: Lab1-Objective1 = Finished
7: ELSE
8: Lab1-Objective1 = Not Finished
9: END
10: End
```

Transportation
The second step addresses a common tradeoff in cybersecurity between performance and security. Rsyslog w/TLS was chosen as the transport mechanism, it provides the equivalent security of other protocols such as SSH. However, it is faster and requires less network bandwidth. Further, it supports an Rsyslog server resides behind a firewall. The Rsyslog traffic is sent over a network separate from where students would perform their lab tasks, as shown in Figure 2.

![Figure 2. Each VM with a score agent has two network connections, the ASMB network is where score data is transmitted and the Lab network is where Lab tasks are performed.](image2)

Storage
In step three, the score data needs to be stored in a secure location, and in a common format. The score format needed to provide five key pieces of information, the student ID number (SID), lab number (LN), objective number (ON), objective state (OS), and if it is attack or defend data (A/D). Figure 3 illustrates our chosen format. For each VM, there is a corresponding log file holding their raw score data. Access controls to the data files provide sufficient data integrity.

![Figure 3. Example of a single round of score data for student2, lab3, objectives1-4.](image3)

Analysis
Step four is where the raw score data is analyzed. The analysis happens as data is received by the Rsyslog server and is broken down into two major steps. The data is normalized and combined into a single file. Then, the SID, LN, ON, OS, and (A/D) are extracted, a database of score and message data is referenced, and the corresponding students’ score and message is updated. Figure 4 and 5 show how the data is interpreted.

![Figure 4. Score data analysis, extracting (A/D), SID, LN, ON, and OS.](image4)

Display
The final step displays would be as simple as populating a table with the students’ username, score and message values stored in the database. As shown in Figure 6.

![Figure 6. Example Scoreboard displaying student scores and messages.](image6)

Materials and Methods
Using PowerShell and Bash scripts, the agents are light weight and require minimal dependency’s making them easy to install on new Lab VMs. Hidden agents, a stringent P/Sense firewall, TLS encrypted traffic and a Security Enhanced Linux Rsyslog Server ensure that score data is handled securely at every stage of the process. Python scripts are used to analyze the data. Score and message definitions and student score and message values are stored in a MongoDB Database, which is easily queried for display by a web application. A MEAN (MongoDB, Express, Angular, Node.js) web stack is used to display the values directly on a Score and Message Board in the CLAaaS learning system GUI.

Acknowledgments
This research is made possible by the National Science Foundation grant 1723650. The authors are grateful to the support of the Department of Technology Systems in the College of Engineering at East Carolina University.