Astrum

by

Vincent Neiheisel, Brett Johnson, & Brenna Martz

Submitted to
the Faculty of the School of Information Technology
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Technology

Vincent Neiheisel _____ 04/06/2021
Vincent Neiheisel Date

Brett Johnson _____ 04/06/2021
Brett Johnson Date

Brenna Martz _____ 04/06/2021
Brenna Martz Date

Ryan Moore, Faculty Advisor _____ 04/19/2021
Ryan Moore, Faculty Advisor Date

University of Cincinnati
College of
Education, Criminal Justice, and Human Services

April 2021

TABLE OF CONTENTS

Table of Contents

TABLE OF FIGURES

ABSTRACT

The Astrum Network Scanner provides small business IT administrators the ability to keep their endpoints secure while saving time and money. According to CenturyLink in 2019, the average cost for network monitoring tools for a business of 100 devices could end up being a pricey $2,656 a year. We have worked towards making an inexpensive and unique solution available. The scanner utilizes Linux-based scripting to scan endpoints and report on any common vulnerabilities as recommended by NIST. This allows IT administrators to automate a large portion of their security auditing. This proved successful through testing during the 2020 – 2021 school year. All this scanning capability is achieved through a one-time deployment of the Astrum solution. Astrum is inexpensive, open-source, and perfect for small businesses on a budget.

1. INTRODUCTION

The following section will describe the overall goals of the Astrum appliance. This includes the problem that the tool solves, more information about that solution, project goals, and a look at the rest of the report.

1.1 Problem

Small businesses and companies often only have a small number of information technology (IT) administrators. These IT administrators oversee all things related to IT and do not have much time to dedicate to network security when they must assist end users with their issues. All networks, big or small, are subject to hackers and other individuals with malicious intent. If the businesses store personal information or credit card information, then that information is valuable. A disgruntled employee can look over their shoulder and see their coworker's passwords stored in a folder. It is the small vulnerabilities that can lead to stolen information and a cyberattack. According to CenturyLink in 2019, the average cost for network monitoring tools for a business of 100 devices could end up being a pricy $2,656 a year (CenturyLink, 2019).

No matter the importance of the data you must protect, whether it be credit cards, personal information, or even first names, it is always recommended to protect your network. One of the first steps that any IT administrator can do is run a network scan. Protecting your network starts with knowledge of what kind of endpoints are communicating on said network. There is a collection of free tools available on the Internet to find out information about a device such as nmap (a tool that identifies endpoints on a network using IP addresses or hostnames) and online vulnerability databases. These tools are beneficial because they can find the information that can help

protect a network, but there is a lengthy series of steps needed to run each set of tools individually. This can be a hassle for the small businesses with one IT administrator available to handle network security.

1.2 Solution

The Astrum tool utilizes the collection of free tools available and provide an all-in-one network visibility tool. This tool follows the NIST cybersecurity framework for small businesses. This framework outlines a five-step process for securing a small business. The team made the Astrum tool focus on the third step of this framework, which is to detect and monitor endpoints on the network (Federal Trade Commission, 2020). This tool, which is also referred to as an appliance, is be hosted on an internal web server and all the hard work is accomplished behind the scenes, allowing an IT administrator more time to dedicate to other projects with their business. Upon entering in an IP address range or endpoint hostname, the Astrum tool runs a collection of scans against the device and finds out as much information as possible. It then uses the information that it found and compares it to a port vulnerability database, as well as report on any other anomalies it located. This can be achieved using user authentication and the use of Linux tools such as curl (a tool that can pull information from web pages) and, as mentioned before, nmap. If the Astrum tool detects an issue with a device, the network analyst can generate a report and pass that information on to whomever would be responsible for managing the device. If possible, the IT administrator would get a custom script to run on the endpoint which could close ports, activate antivirus software, deactivate usb ports, and contain links to advice on where to find out more information about vulnerabilities.

The purpose is to provide a free solution for a convenience tool that directly protects all users within a business or organization. The information returned is about real-world device vulnerabilities. This provides a way for small business IT administrators to save time and focus on all their responsibilities. An ease-of-access tool such as this may provide the motivation to help protect small networks and the devices on it. More importantly, the users on the network can feel more confident in the security of their devices.

1.3 Project Goals

The goal of Astrum is to provide an inexpensive ease-of-use tool for small business IT administrators. The tool will be able to be used on Linux based operating systems with minimal hardware requirements. Linux acts as the operating system because it can run scripts in the bash shell, which is the core language that Astrum uses. During operation, once options are selected on a web interface, the Astrum tool will scan for vulnerabilities in a network. Once the scan is done, the IT administrators can utilize the tool to remedy said vulnerabilities.

1.4 Overview

The remainder of this final report will discuss in detail how this project was completed. This includes sections covering the project concept, objectives, methodologies, user profile, use case, problems encountered, budget, timeline, and the conclusion.

2. DISCUSSION

In this section we discuss the technical aspects of the Astrum appliance, especially those accessed during user acceptance testing, as well as the project management items including budget, cost, and time. We will also discuss problems encountered as well as recommendations for improvement.

2.1 Project Concept

The project team was inspired by Linux tools such as nmap, curl, dig, and the universal tool on many operating systems; nslookup. With these tools in mind, the group wanted to combine the usefulness of these tools and develop something that could be used in real-world environments as a cheaper solution. Brenna initially pitched the idea for a scanning tool that would utilize some of these tools to help with security analysis. Vincent took that idea and suggested a report feature, and then Brett went to implement a web server-based idea.

During the summer semester of 2020, the group met multiple times via WebEx to discuss the project's necessities and finalized the concept of the Astrum tool. This would be a tool that would assist busy IT administrators to scan for vulnerabilities and provide network insight. The group then worked to create a project timeline with the goals described below in mind. Once this was complete, development began with the setup of the testing environment and network.

2.2 Design Objectives

The goal of the project was to successfully develop a miniature security scanner that will be primarily used by IT administrators that oversee every IT-related responsibility in a network. There is a web interface hosted on a web server that allows

for remote access. The server must be deployed on a Linux operating system on an internal network. The sever hosts a web interface and is able to generate reports and remediations on an endpoint if desired.

The Astrum tool is Linux-based and runs scripts that allow for ease of information collecting on an endpoint. The scripts that run in the background are primarily written in bash and perform various functions such as nmap and remote SSH logins. The information retrieved is then compared to a remote vulnerability database and checks are run on potential issues with systems. It then provides any possible solutions for an IT administrator via a resolution script that provides what is described in the solution as a script that can "close ports, activate antivirus software, deactivate usb ports, and contain links to advice on where to find out more information about vulnerabilities."

2.3 Methodology

Web Interface

The web interface runs on a Linux-based web server using technologies like Node.js, Express.js and Pug.js. This was chosen over Apache due to the server-side scripting ease. The overall goal was to have the web interface house the main script that the Astrum appliance runs. One of the original concerns the project team had was with the free-to-use tools being a long series of steps to run individually. The "single pane of glass" interface was to enforce the ease-of-use goal that this tool provides.

Open-Source Tools

The Astrum team chose to use open-source tools only to ensure that the appliance's code remained free. These tools include nmap, bash, curl, openSSH, and sshpass. The real usefulness of the Astrum appliance comes from the organized scripting

that makes use of these tools to remote into endpoints and gather data. Keeping things secure, this requires a user login that the IT administrator can set up via active directory (AD) or locally on the machine.

Hardware and Other Considerations

The only possible costs with using Astrum comes with purchasing hardware to run the web server. As discovered during development, an old computer with minimal hardware specs can be used to run Astrum. The deployment comes in a set of instructions available on GitHub, linked in the appendix, to install all of Astrum's dependencies and is similar to other open-source tools' installation process.

2.4 User Profile

Astrum is a server-based application that handles security scanning for small business IT administrators. That is our main set of users. Secondary users would be employees at the small business that receive reports and scripts that they need to run on their computer to resolve vulnerabilities.

Potential users – Small business IT administrators

Software and Interface Experience: The user should know how to set up a Linux server, as well as check for updates on occasion. They should also know how to navigate a basic web GUI interface in order to run the application.
Experience with Similar Applications: The user should have experience with server deployment; VMware or physical deployment. Basic Linux CLI and GitHub knowledge needed to update application.

Task Experience: The user must be able to check for updates and use a web interface. The user must also have a procedure for remedying vulnerabilities after using the application. This would involve tasks like e-mailing and document creation.
Frequency of Use: This is up to the discretion of the user. Depending on the extent of the scan, the user could run this multiple times a day if they wanted to. A small business should not scan that frequently, but it is possible.
Key Interface Design Requirements that the Profile Suggests: The user will need to be able to navigate and use a web interface. They need to be able to follow instructions on GitHub to get the appropriate files on a Linux server using the CLI.

Potential Users – Small business employees

Software and Interface Experience: The user must be able to use a computer enough to click and type objects. They must be able to follow a set of instructions that include Images.
Experience with Similar Applications: Outlook, office tools, basic web browsing. Preferably familiarity with the Terminal window of their respective operating system.
Task Experience: Following a set of instructions including pictures. Ability to check e-mail and follow deadlines. Knowledge of how to contact IT support.
Frequency of Use: They will need to be able to run the remedy script as often as their IT administrator suggests that they do so. The IT administrator will be responsible for sending notifications to the employee end user that will need to make changes to their machine.

Key Interface Design Requirements that the Profile Suggests:

This will primarily be left for the IT administrator to decide how they want to deploy the remedy scripts that fix any vulnerabilities located. Astrum will suggest methods to the IT administrator in the README.

2.5 Use Case Diagram

Below displays the use case diagram for operation of the Astrum appliance. The two users would be a small business IT administrator and a small business employee. The use case displays how the Linux server running Astrum does the hard work in the background, and the functionality is controlled using the web interface.

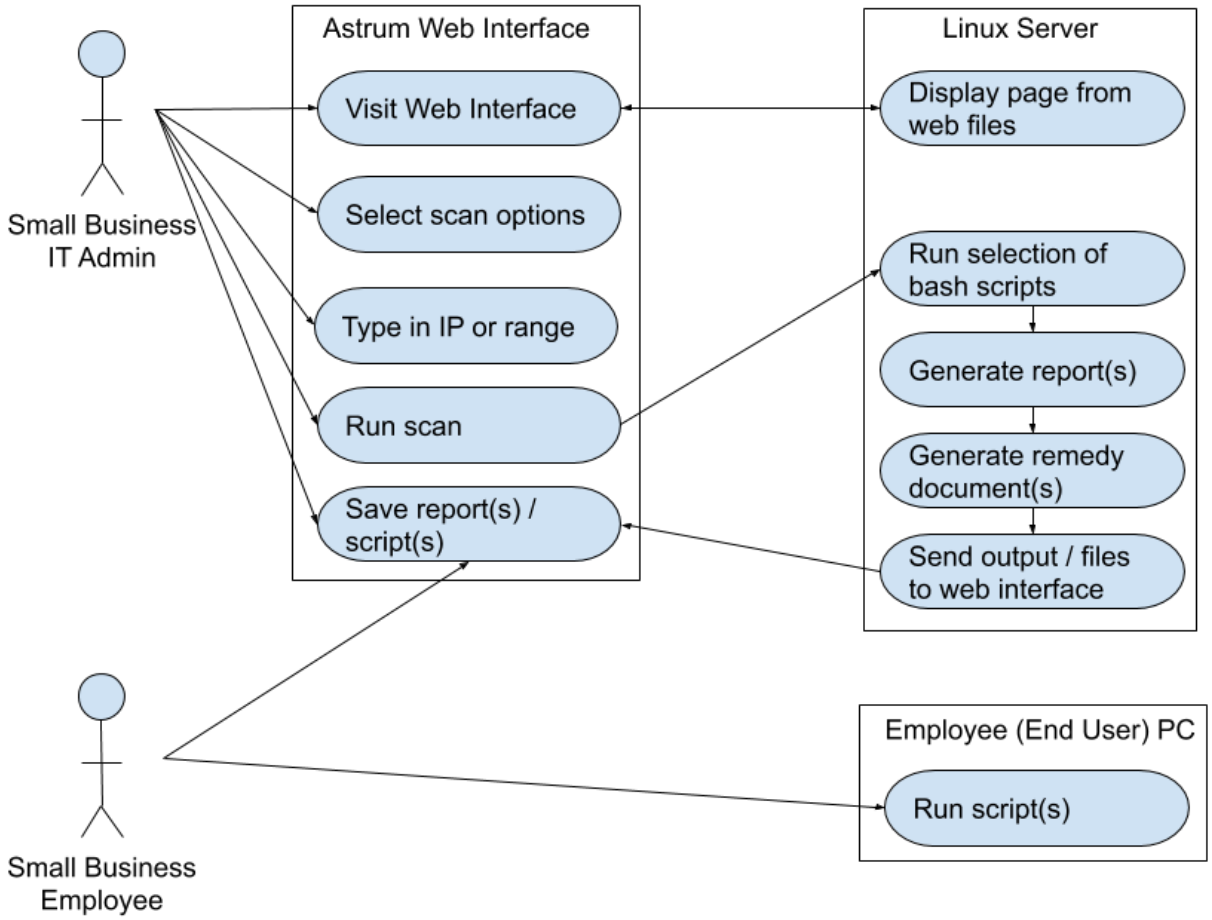

Figure 1 Use Case

2.6 Technical Architecture

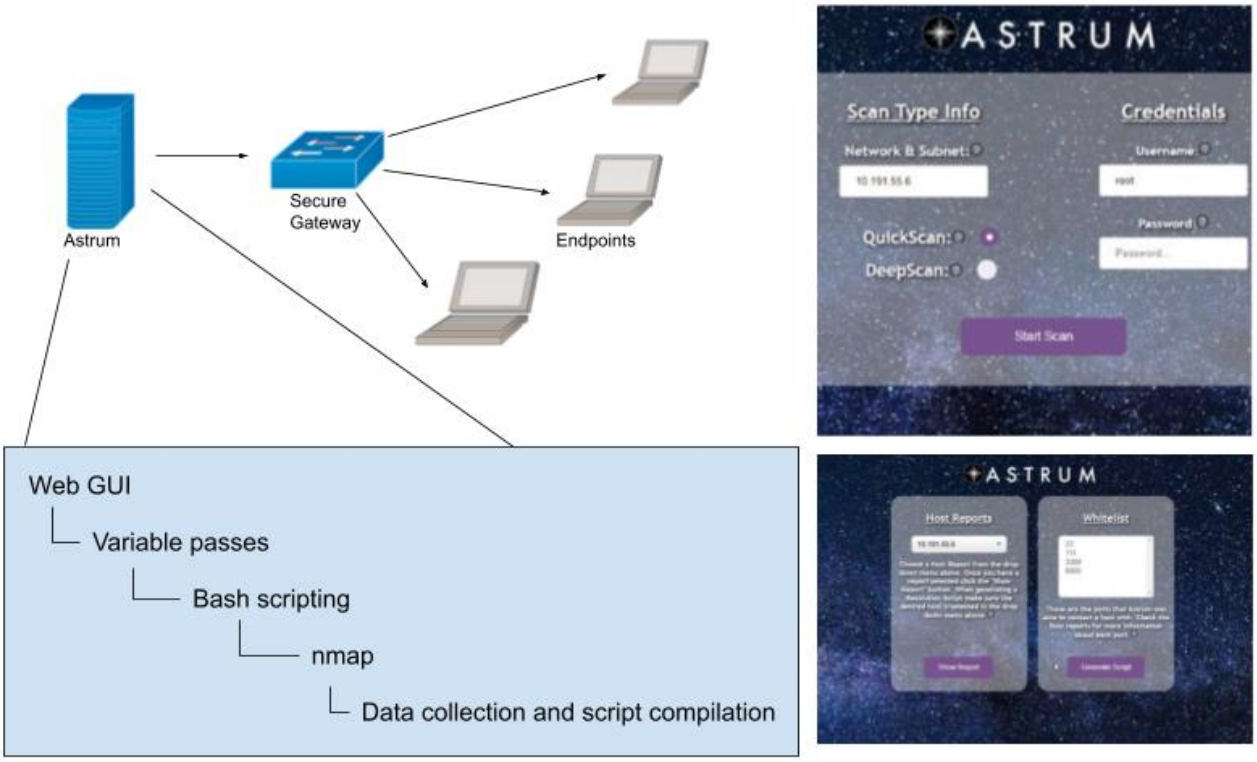

Figure 2 Technical Architecture

Astrum runs on a Linux web server utilizing node.js and express.js to run its web capabilities. It is recommended that the server is placed in a secure environment and access lists permit / deny traffic at the network level. The use case section above displays the series of events that the server runs in order to complete a scan. Aside from the web server necessities, Astrum runs completely on bash and windows command shell scripting. Information is gathered using credentialed ssh logins. If Astrum can log in to the endpoint, it will begin the appropriate bash / windows command shell scripting. If Astrum can not log in, it will still report on the endpoint on the results page, but the information will be limited. This still provides network visibility and may prompt for investigation depending on what OS the nmap scan detects.

2.7 Testing

In this section we will review the methodologies used for testing Astrum, the scope of testing, the objectives intended to reach, test logs and procedures, and a summary of what we learned throughout the testing period. These results represent Astrum as it was in its original completed state.

Testing Methodology

How:

After Astrum is set up, there is not much for the end user to test. The aspect we want to focus on verifying is the backend scripting and its stability. This means we need to make sure variables were being passed correctly from the user's input on the web interface to the Linux server for scripting. These variables determine what the script will do, so that many different combinations are assessed. These combinations lead to different devices, primarily a variety of Linux and Windows hosts, being scanned and evaluated for runtime errors and other bugs.

Why:

Astrum is meant to be used primarily by small business IT administrators. When we have users test this product, we need to ensure that they are in the IT field and would have enough knowledge to already know what a subnet is and be able to type it in to the web interface fields.

Scope of Testing

The elements that will be covered are the web interface stability, the server-side scripting, and the effectiveness of the resolution scripts on hosts that have been deemed

to have vulnerability issues. The specific subnet being scanned does not matter so long as the hosts on the subnet change during testing.

For web stability, we need to have IT employees type in appropriate information successfully. This information will be run in a test environment, so their choice of subnet will be limited. Once they run the scan, we can watch the backend console to make sure that the variables were passed correctly. The reports that are generated will be assessed to see if the proper information was gathered. In a test environment, we can intentionally set up vulnerabilities on Linux and Windows machines which shows us if Astrum was able to discover them. After the reports are generated, the IT users testing Astrum will select ports to whitelist and run the resolution script. We will run these ourselves since the user will not have access to the hosts due to the COVID-19 restrictions. This will be a rinse and repeat process with different hosts and vulnerabilities.

Objectives

We are looking to answer the following questions based on the use case flow:

- Can the user type in the information they want in the text boxes?
- Will Astrum pass the variables correctly, and if the variables are invalid will it properly error out?
- Will Astrum call the proper scripts and run remote scans?
- Are the reports generated by Astrum accurate and formatted properly?
- Can the user select these reports, analyze them, and select the ports they wish to whitelist?
- Will Astrum provide the resolution / remedy script for Windows and / Linux devices?
- Can the Astrum team run these scripts on hosts and patch vulnerabilities on remote hosts?

All of these should be answered with a yes before the IT Expo on April 13th, 2021. Due to COVID-19 restrictions, the Astrum team will play the role of the small business

employee regarding running the remedy script on hosts since any users testing Astrum will be unable to access these devices.

As of early March 2021, multiple sessions of testing have revealed led to improvements that are defined in the testing review section. The objectives tested are listed below as well as their results at the time of testing.

Test Logs and Procedures

The pass conditions include answering yes to the questions listed in the objective section, and the failure conditions include answering no as well as any negative feedback from users during testing.

TEST ONE

User / Role	Objective Tested	Result	Notes	Date / Time Tested
Rob N. / Jr. Network Administrator	Are the users restricted to only valid entries in the form fields?	No	• Invalid hosts lead to blank results page. • Blank fields lead to node.js/JavaScript error page.	02/12/2021 09:15AM
Rob N. / Jr. Network Administrator	Does Astrum pass variables correctly?	Yes		02/12/2021 09:15AM
Rob N. / Jr. Network Administrator	Does Astrum properly error out when bad variables are passed to it?	No	• More work to do here.	02/12/2021 09:15AM
Rob N. / Jr. Network Administrator	Does Astrum call the proper scripts?	Yes		02/12/2021 09:15AM
Rob N. / Jr. Network Administrator	Are reports generated by astrum accurate and formatted properly?	Marginally good	• Port list needs formatting. List did not wrap.	02/12/2021 09:15AM
Rob N. / Jr. Network	Can the user select the ports	Yes		02/12/2021 09:15AM

Administrator	they wish to whitelist and generate a script?			
Rob N. / Jr. Network Administrator	Can Astrum provide scripts for Windows & Linux devices?	Marginally Good	• Linux scripts generated but need improvement. • Unsure about Windows scripts.	02/12/2021 09:15AM

Question	**Comments**
What do you think of the interface? Is there anything you like/dislike?	• Navigation buttons are needed, there is some uncertainty when using the browsers navigation. (Forward, Back, etc.) • User expects the logo to be a link to the main page, but it is not. • User expects the underlined headings in the reports to be links, but they are not. • Issues with how the help tips are overlayed creates confusion.
What do you think of the overall user experience?	• "Look and feel is good."

TEST TWO

User / Role	**Objective Tested**	**Result**	**Notes**	**Date / Time Tested**
Astrum Team / Developers	Are the users restricted to only valid entries in the form fields?	Yes	• HTML5 combined with regex was implemented to fix this.	02/22/2021 07:13PM
Astrum Team / Developers	Does Astrum call the proper scripts and run remote scans?	Yes	• Console output and reports show correct information	02/22/2021 07:13PM
Astrum Team / Developers	Can the user select these reports, analyze them, and select the ports they wish to	Yes		02/22/2021 07:13PM

	whitelist?			
Astrum Team / Developers	Can Astrum provide scripts for Windows & Linux devices?	Yes	• While the generated script is accurate – there are some miscellaneous lines of code. Does not impact functionality.	02/22/2021 07:13PM
Astrum Team / Developers	Can the Astrum team run these scripts on hosts and patch vulnerabilities on remote hosts?	Marginally Good	• Varies on host machine configuration – works on fresh OS installs	02/22/2021 07:13PM

Testing Review

We have learned during continued testing that bugs are unpredictable. We also learned that user acceptance testing should be done earlier in the process so we can receive appropriate feedback. The most obvious thing that users were able to catch that we did not originally have was a loading page.

Utilizing the use case to generate items for users to test was effective. We should have had users look at Astrum during development phases to catch the obvious additions that greatly enhanced the appliance. The project utilized more of a waterfall methodology and we would have liked to adapt the agile methodology overall.

2.8 Budget

The figures below display the budgeting considerations the Astrum team had to make during development. As promised, the actual code remains open source and free,

and the only potential cost comes from the hardware that an IT administrator chooses to run the code on.

Project Name:	Astrum	**Project Manager:**	Brenna Martz	**Team #**	24
Project Members:	Vincent Neiheisel, Brett Johnson, Brenna Martz	**Project Areas:**	Cybersecurity	**Project Advisor:**	Ryan Moore

Problem Statement

Small businesses and companies often only have a small number of IT admins. These IT admins oversee all things related to IT and do not have much time to dedicate to network security when they must assist end users with their issues. All networks, big or small, are subject to hackers and other individuals with malicious intent. According to CenturyLink in 2019, the average cost for network monitoring tools for a business of 100 devices could end up being a pricy $2,656 a year (*Weighing the Price of Security* . CenturyLink, 2019).

No matter the importance of the data you must protect, it is always recommended to protect your network. One of the first steps that any IT admin can do is run a network scan. Protecting your network starts with knowledge of what kind of endpoints are communicating on said network. There is a collection of free tools out there to find out information about a device such as "nmap" and online vulnerability databases. These tools are nice, but it is a lengthy series of steps needed to find information about a device on a network. This can be a hassle for organizations and businesses that have one or two network analysists.

Project Description

The Astrum tool will utilize the collection of free tools available and provide an all-in-one network visibility tool. We want this tool to follow the NIST cybersecurity framework for small businesses. This framework outlines a five-step process for securing a small business. We want to focus Astrum on the third step of this framework, which is to detect and monitor endpoints on the network (*Cybersecurity for Small Business* . Federal Trade Commission & NIST, 2020). This tool will be hosted on an internal web server and all the hard work will be accomplished behind the scenes, allowing an IT admin more time to dedicate to other projects with their business. Upon entering in an IP address range or endpoint hostname, the Astrum tool will run a collection of scans against the device and find out as much information as possible. It will then use the information that it found and compare it to a port vulnerability database, as well as report on any other anomalies it located. This can be achieved using user authentication and the use of Linux tools such as curl and nmap. If the Astrum tool detects an issue with a device, the network analyst can generate a report and pass that information on to whomever would be responsible for managing the device. If possible, the IT admin would get a custom script to run on the endpoint which could close ports, activate antivirus software, and more.
Overall, the purpose is to provide a free solution for a convenience tool that directly protects all users within a business or organization. The information returned is about real-world device vulnerabilities. This provides a way for small business IT admins to save time and focus on all their responsibilities. An ease-of-access tool such as this may provide the motivation to help protect small networks and the devices on it. More importantly, the users on the network can feel more confident in the security of their devices.

Project Asset Type

Compliance/Regulatory	Comments: The Astrum appliance will assist with NIST guidelines for small businesses. It will also note vulnerability information that is current reagrding ports.

Funding Source (if applicable)

Self	Comments: We are providing our own hardware and network access for this project.

Risk Identification (See Risk Types tab)

	Risk Rating 1-5 (5 is high)*	*Comments*	*Weight*	*Score*
Work Effort (days)	3	Open-source but put togther.	40%	1.20
Complexity	3	One time set up.	60%	1.80
Project Risk Score:				**3.00**

Project Stakeholder(s)

Small business IT administrators, smaill business technology users, Astrum developers.

Figure 3 Budget Description, Source, and Risk

Estimate of Benefits

If project will generate revenue, estimate 1 year here:	$ -

Select other benefits the project may bring a customer or user:

Risk Avoidance	☑
Improved customer satisfaction	☐
Increased system availability	☐
Productivity or process improvement	☑
Reduced costs	☑

Estimated Cost Rough Order of Magnitude:

	Rate Per/Hr	Work Effort (Hours)	1 X Costs	Ongoing Annual Rate Per/Hr	Ongoing Annual Work Effort (Hours)	Ongoing Annual 1 X Support Cost
Labor - IT	20	270	$ 5,400.00	20	80	$ 1,600.00
Labor - External	0	0	$ -	0	0	$ -
Software - External			$ -			$ -
Hardware - External			$ 500.00			$ -
Misc.			$ 750.00			$ -
TOTAL			$ 6,650.00			$ 1,600.00

Comments: The 1x costs relate to the hardware that Astrum needs to run on, of course this can vary, so $500 is a good mid-range computer that it can run on.The miscellaneous costs relate to any networking cables or switch connections that Astrum will need to communicate with the network. Of course, this cost could be zero assuming the company already has hardware and a network to use.

5-Year ROI Analysis

Description	5- Year Expected	Conservative (1.5)
Total Costs	$ 14,650.00	$ 21,975.00
Total Benefit	$ -	$0
Total Costs/Benefit Differential	$ (14,650.00)	
Conservative Costs/Benefit Differential	$ (21,975.00)	

Figure 4 Budget Benefits and Costs

2.9 Project Timeline

Below shows the dates that the team followed in order to complete the Astrum appliance. Many of the dates revolved around having the alpha completed by the end of 2020. The spring focuses on design of the web interface and preparation for the final product.

Task Name	Duration	Start	Finish
Senior Design Project	**172 days**	**Mon 8/17/20**	**Tue 4/13/21**
Problem & Project Description	11 days	Mon 8/17/20	Mon 8/31/20

Team Contract	16 days	Mon 8/31/20	Mon 9/21/20
Project Abstract	30 days	Tue 9/1/20	Mon 10/12/20
Team Contract Resubmission	15 days	Tue 9/22/20	Mon 10/12/20
Web Server	**25 days**	**Mon 8/31/20**	**Sun 10/4/20**
Initial configuration	11 days	Mon 8/31/20	Sun 9/13/20
Script Research and Modifications	11 days	Mon 9/14/20	Sun 9/27/20
New Feature Implementation & Testing	6 days	Mon 9/28/20	Sun 10/4/20
Scanning Script	**77 days**	**Sun 8/23/20**	**Tue 12/8/20**
Research/Design/Testing	78 days	Sun 8/23/20	Tue 12/8/20
Week 9 (3 Assignments Due)	**10 days**	**Mon 10/5/20**	**Sun 10/18/20**
Elevator Speech	11 days	Mon 10/5/20	Sun 10/18/20
Use Case Diagram	11 days	Mon 10/5/20	Sun 10/18/20
User Profile Report	11 days	Mon 10/5/20	Sun 10/18/20
Generated Scripts	**67 days**	**Sun 9/6/20**	**Tue 12/8/20**
Research Generated Script	68 days	Sun 9/6/20	Tue 12/8/20
Design Reports	58 days	Sun 9/20/20	Tue 12/8/20
Testing Generated Script Functionality	27 days	Mon 11/2/20	Tue 12/8/20
Vulnerability Reporting	**20 days**	**Wed 11/11/20**	**Tue 12/8/20**
Research/Design/Testing	20 days	Wed 11/11/20	Tue 12/8/20
Fall Report Draft	11 days	Mon 11/2/20	Sun 11/15/20
Final Fall Semester Report	11 days	Mon 11/16/20	Mon 11/30/20
Final Fall Oral Presentation (8 min)	0 days	Mon 11/30/20	Mon 11/30/20
Alpha Project Complete, last day of Semester	1 day	Tue 12/8/20	Tue 12/8/20
Winter Break	24 days	Wed 12/9/20	Sun 1/10/21

Spring Break	6 days	Mon 3/15/21	Sun 3/21/21
Web/Dashboard Design	**15 days**	**Sun 10/18/20**	**Fri 11/6/20**
Design	11 days	Sun 10/18/20	Fri 10/30/20
Testing	11 days	Sun 10/25/20	Fri 11/6/20
Final Testing	**67 days**	**Mon 1/11/21**	**Tue 4/13/21**
IT Expo	1 day	Tue 4/13/21	Tue 4/13/21

2.10 Problems Encountered and Analysis of Problems Solved

Development of applications comes with potential roadblocks and new solutions. The main problem the Astrum team encountered with Astrum was early in the development. They recognized that there was no way to easily obtain the information they were looking for to help comply with the NIST Cybersecurity Standard for small businesses. The solution was found by using sshpass, which allows for remote terminals into machines using a login for that machine. On Linux devices that resulted in a shell terminal to gather information, and on Windows devices that allowed for access to a command prompt terminal for information.

Remoting into Windows devices was also another problem that was solved by enabling openSSH on Windows devices. On Windows 10, they discovered that it was bundled with the operating system and only needed to be enabled. With it enabled, port 22 was opened on Windows devices and remained secure.

The second biggest problem showed itself during the second half of development when the web GUI was being implemented. The translation between standard web HTML and CSS into pug code for Express.js proved to be not as simple as we thought.

There were consistent small errors with hover elements, padding, and logo displays. The solutions for these involved sitting and making minor adjustments to first the CSS and then the pug code to see if we could remedy these items.

2.11 Recommendations for Improvement

We would have liked to approach the agile development cycle earlier on in the project. The course is very much scheduled to suit a waterfall method of development. We received a large amount of very useful feedback from users too late in the project to truly optimize its potential. We were still able to add items that users recommended, such as a loading screen and tooltips. If we had more time, we would have liked to add navigation buttons and other visual enhancements. Astrum will remain open source and available for anyone that wishes to utilize it as of April of 2021.

3. CONCLUSION

This section discusses the Astrum team's skillset that was utilized for this appliance. It also briefly discusses what skills they were able to build upon and what they plan to do for the beginning half of 2021.

3.1 Lessons Learned So Far

The main lesson learned for everyone involved in the development process is that not everything you can think of is possible or easy to accomplish. Since the team is only made up of students with minimal experience in the real-world, it was hard to work with what they knew. If this project were to be handled by a team of development experts, then they could assume that there would be a proprietary tool developed to better handle the scanning process.

3.2 Abilities Developed Throughout Project

Thanks to the classes that they have participated in before, the project management aspect of this appliance was not difficult; most new skills came in the form of code development. Operating system specific terminal commands and overall Linux server set up and management skills were enhanced.

Brett Johnson particularly enhanced his skills in node.js and express.js tools as he took a big lead in the web interface development.

REFERENCES

CenturyLink. “Weighing the Price of Security”

2019. Accessed September 24, 2020. www.centurylink.com/asset/business/enterprise/guide/weighing-the-price-of-security-cost-comparison-guide.pdf.

Federal Trade Commission & NIST. “Cybersecurity for Small Business”

2020. Accessed September 29, 2020.

www.ftc.gov/system/files/attachments/understanding-nist-cybersecurity-framework/cybersecurity_sb_nist-cyber-framework.pdf.

APPENDIX

Appendix A. Project Code

The link to the GitHub with the code for Astrum can be found here: https://github.com/Astrum-Group24/Astrum