NanoFirewall
Design Document

A portable network traffic analyzer designed for a single-board computer

---

Spencer Cameron
Evan Hirn
Tyler Klein
Hunter Moffat

Prepared for CS 4000
University of Utah

Table of contents

---

1. Executive Summary

Billions of people around the world access the Internet daily. For many unprotected networks, this means you are vulnerable to exploits and attacks from across the globe. These attacks could happen while you are out getting coffee or even in the comfort of your own home. Furthermore, as smart devices and Internet-based home security systems gain popularity, these everyday devices become an entrypoint for hackers into our homes. The average home network owner does not have access to the technical knowledge and expensive equipment required to set up their own security. NanoFirewall is a portable network security device that presents a middle ground for people who can't justify a large-scale firewall but still value their Internet safety. This product not only benefits individual users, but small businesses looking for a robust and affordable security option.

The product itself is a tiny computer, about the size of your palm, that plugs directly into a network to provide security by analyzing the packets that pass through. The NanoFirewall can be placed in the path between your home network and the outside world. Alternatively, the device can be positioned between your personal laptop and an unsecured public network, such as a coffee shop. By doing this, our firewall is able to watch out for a variety of user-designated rules. While in a firewall mode, the device can stop the packet dead in its tracks. Alternatively, the device can be put in a reporting mode where a triggered rule instead notifies the network administrator. Some of these features include, but aren't limited to, blocking different devices or websites, restricting connections to foreign countries, ad block, virus detection, parental controls, and more. Our firewall will be configured by an easy-to-use web interface that is hosted on the device itself, allowing easy management anywhere on the network. For those who aren't as tech savvy, the firewall comes preset with safe default rules that will improve your home security.

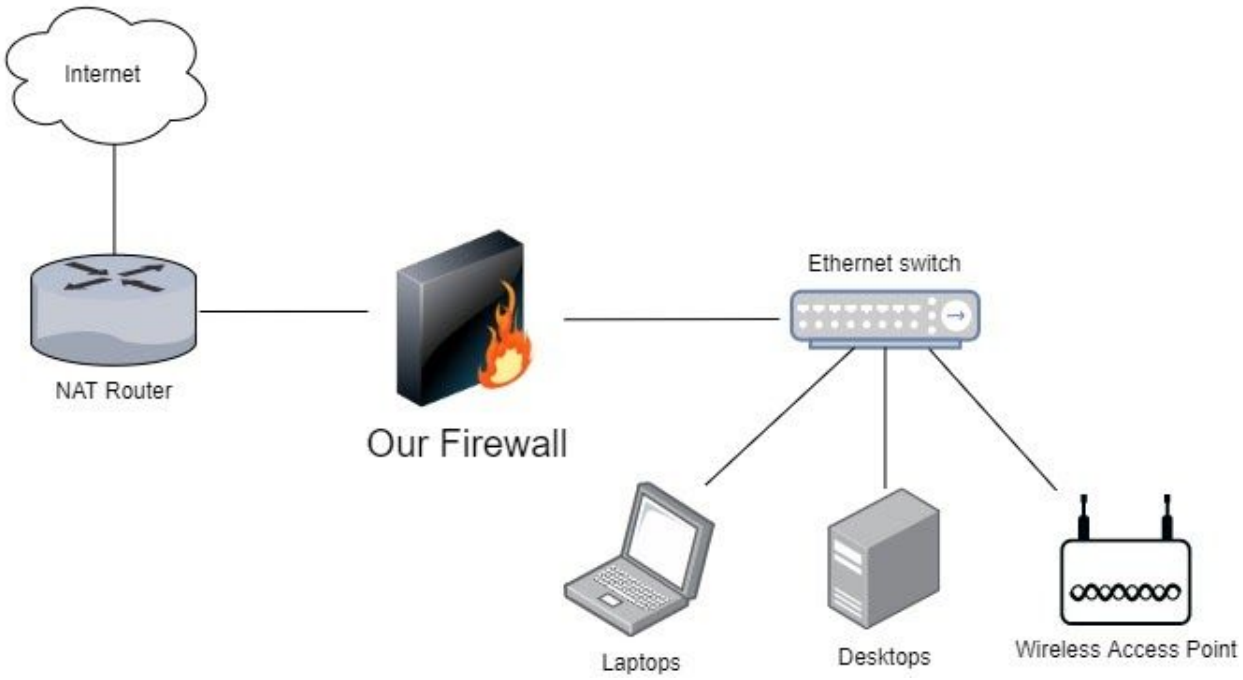


Figure 1: NanoFirewall Overview

2. Background

2.1. Idea Space

Only 4% of people in the United states thoroughly understand firewalls, and 44% don't even know what one is [1]. However, these people still need to protect their homes and businesses. Firewalls can be difficult to set up properly for the average user. When users want a firewall, their only options are pricey, overly-complicated hardware that are meant for large-scale systems and corporations. Many small businesses and everyday users aren't able to purchase or use such a robust firewall. This leaves them vulnerable with few security options.

Our solution to this problem is to provide users with a simple and easy-to-use firewall. This firewall fits in the palm of your hand and gives you more control of your local network and Internet traffic. The firewall will be situated between two points in the network and can block/monitor traffic moving through. It will come preloaded with security defaults to give peace of mind and help those who may not have the technical knowledge required to set up their own network rules. In addition to this, we will also include a user-friendly interface for more advanced control over your network. The intended users of this system are people who may not have the skill level, time, or money required to set up a full-scale firewall for their home network or business. That is to say, most users will require basic security, but don't need an advanced system.

2.2. Similar Ideas

Firewalls are certainly not a novel idea, and have been around since the conception of computer networking and the Internet. While Cisco dominates the enterprise firewall market, they are not the competitor of NanoFirewall. Instead, NanoFirewall is designed for home networks and small businesses. We will focus on three similar implementations to our product: Firewalla Blue, Bitdefender Box, and PiHole.

2.2.1. Firewalla Blue (https://firewalla.com/products/firewalla-blue)

Firewalla is a company which makes affordable cyber security appliances, and their blue model is most similar to our product. The Firewalla and our NanoFirewall differ in numerous aspects. The most prominent difference is that the Firewalla only has a single Ethernet interface to connect into the network. It uses clever hacks to make the firewall work on a single port, such as ARP spoofing or multiple VLANs. Our NanoFirewall comes in two different options, but both work as a link between two Ethernet interfaces, offering a hardware-level disconnect that is much more difficult to bypass. Another con of Firewalla is that it requires a compatible router. Furthermore, Firewalla uses a mobile phone application for settings management whereas our NanoFirewall will use a LAN web application, which does not restrict what type of devices can use our firewall manager.

2.2.2. Bitdefender BOX (https://www.bitdefender.com/box/)

The Bitdefender BOX is a standalone firewall that plugs into your router or home gateway. The obvious difference between the Bitdefender BOX and the NanoFirewall is that the Bitdefender BOX relies on a subscription service, as well as the initial hardware purchase. The $150 initial purchase in conjunction with the $99/year subscription (after the first year) results in an expensive investment for the average customer who just wants better home security. The NanoFirewall does not need a subscription and only requires a one-time purchase. Just like the Firewalla, the Bitdefender BOX uses a mobile application for settings management. As said previously, the NanoFirewall uses a LAN web application which is not device specific.

2.2.3. PiHole (https://pi-hole.net/)

PiHole is a free and open source DNS monitoring and blocking application that can be installed on a Raspberry Pi. PiHole provides users with basic DNS filtering and statistics. Some inspiration for our UI will be taken from PiHole because of its sleek and informative web interface. Compared to our firewall, PiHole is less consumer friendly by requiring the user to manually configure each device to point to the Raspberry Pi as their DNS server. NanoFirewall is designed with the average consumer in mind, only requiring the user to take it out and

plug it in for it to work. NanoFirewall also includes many more network management features outside of DNS filtering, such as setting data limits for specific devices and setting roles and permissions for different users.

2.3. Required Technology

The system level of our firewall will be programmed in C++ using libtins, a packet crafting and sniffing library [2]. Libtins will be used to implement all of the core features of our firewall. Libtins is an extremely useful library for providing the packet data to our program, but we will be handling the inspection of those packets for rule violations ourselves. Our firewall will also likely make use of the VirusTotal REST API, which can be used to determine notoriety of domains and IP addresses, as well as scan data for viruses. Our application will also use the restbed library for hosting a REST API as well as Nginx for our primary web server. The configuration interface of our firewall is a local website that will be programmed using the React JavaScript framework.

2.4. Software/Hardware Requirements

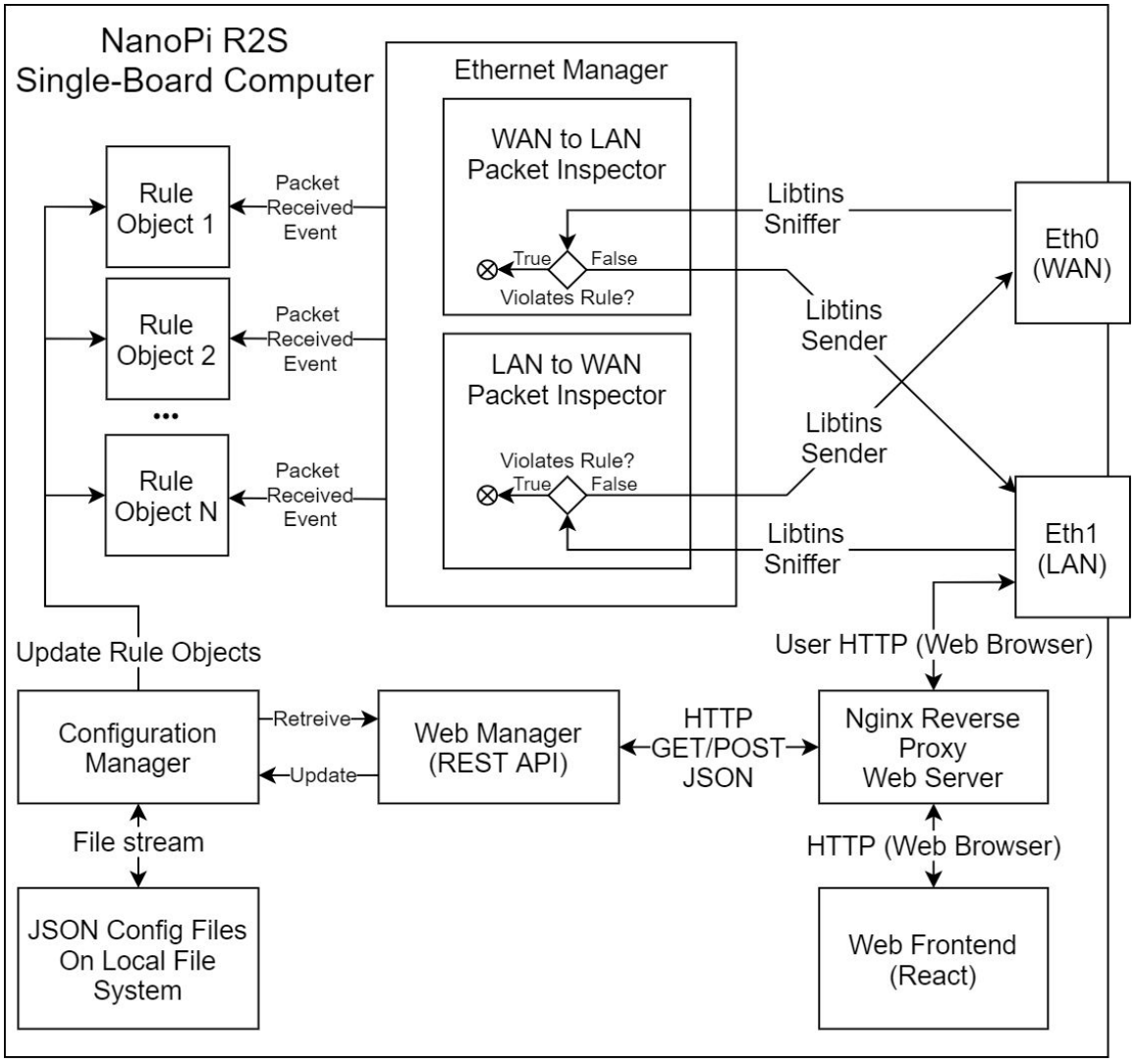
Our firewall will be designed for single-board computers. On the hardware side, we have two separate versions we will be working with. For our wired version, we will be utilizing the NanoPi R2S. The NanoPi R2S houses two separate Ethernet ports, one to connect to your network router (WAN) and one to connect to your local network (LAN). For our wireless version, we are going to be using a RaspberryPi 3, equipped with a USB wireless access point and an Ethernet port. The Ethernet port will connect to your network router (WAN) and the USB dongle will enable the user to connect their wireless devices. Our software will be officially supported by these two hardware configurations, however it may be compilable on other Debian devices without guarantees. The only requirement to actually use our product is a network for it to hook into. A web browser will be needed to access our web frontend.

3. Requirements Analysis

3.1. System Architecture

The overall system architecture of the NanoFirewall device can be described by the Model-View-Controller (MVC) design pattern. The view is made up of the web frontend used for managing the device. The model is the Ethernet Manager used to monitor packets passing through the device. Finally, the controller is made up of the Configuration Manager and Web Manager. This overall architecture can be seen in Figure 2 below.

Figure 2: System Architecture



The model and controller of the firewall run from a single C++ console application. Within this process, two threads are used to read in packets and either forward or block them. Meanwhile, a third thread is used to host the web backend which is a REST API created with the C++ restbed library.

The view, or web management interface, will consist of a React frontend for providing the user with a UI. The web frontend will trigger HTTP GET/POST requests to the REST API running within the C++ application.

The model takes advantage of the Observer design pattern.

- The Ethernet Manager is the subject.
- Each rule object acts as an observer, or subscriber, to an event which is triggered each time a packet arrives.
- The Ethernet Manager has two Packet Inspectors, which use the libtins packet sniffing library in order to read in all the packets on a given Ethernet interface and trigger the event.
- Once a rule object receives a packet, it responds to the monitoring thread with a boolean of whether or not to block that packet. After the primary monitoring thread has received all its responses from subscribed rules objects, it either forwards the packet (using a libtins sender) or drops it.
- If a rule is disabled, it simply “unsubscribes” from this event.

The controller handles retrievals and updates to the configurations

- The Configuration Manager uses the Singleton design pattern.
- It updates rule objects based on JSON provided from configuration files on the local file system or coming from the Web Manager.
- The Configuration Manager also updates the local configuration files.
- It also provides getters/setters to various configuration data.
- The Web Manager handles the HTTP GET and POST requests coming from the users.
- A GET request will provide JSON about the current configuration for a given rule.
- A POST request will use JSON to update the configuration for a given rule.

3.2. Personnel

There are two clearly distinct modules of this product: the firewall system process and the web management interface. Due to this natural division, the personnel will be split into two subgroups. Evan Hirn and Tyler Klein will lead the system programming, while Spencer Cameron and Hunter Moffat will be responsible for the web management system.

Tyler's role in the device is the Ethernet packet monitoring using the libtins library and inspection of each packet using rule objects. He will also be the peer reviewer for Evan's code. Tyler has taken Computer Networks and has a strong background in embedded systems and various networking protocols from industry experience.

Evan will be responsible for the configuration manager. After the configuration manager has a strong foundation, he will head up device statistics and assist in writing Rule Objects. Evan has taken both Computer Networks and Network Security. He has additionally been using C++ at work for the last six months on top of his university experience. Evan will also be responsible for peer reviewing Tyler's code.

Hunter will design and implement the look and feel of the web interface frontend. Hunter has a background in user experience from designing an experiment management system (EMS) for the POWDER 5G testbed this past summer. The entire goal of his EMS was to make deploying, monitoring, and managing experiments on the testbed as easy and possible. He has also taken courses like Mobile App Development and Web Software Architecture giving him a background in web development and user interfaces. He will also be in charge of peer reviewing Spencer's code.

Spencer will be in charge of the backend for the web application, also called the Web Manager. This entails building up GET responses and reading in POST information from the user. Spencer will be working closely with Hunter to insure interoperability between the front and back ends and is responsible for peer reviewing his code. Spencer has a background in coding servers for software practice.

3.3. System Features

Feature	**Rank**
Allow the user to filter IP addresses.	Bare Essentials (1)
Allow the user to filter DNS requests.	Bare Essentials (1)
Allow the user to filter MAC addresses.	Bare Essentials (1)
Allow the user to filter TCP or UDP ports.	Bare Essentials (1)
Allow the user to decide between either a block and allow list for different filters.	Bare Essentials (1)
An intuitive web interface for the user to control their device settings.	Bare Essentials (1)
A login screen to access the management interface with a password	Bare Essentials (1)
Ability to enable/disable the firewall	Bare Essentials (1)
Email notifications to notify the admin if a rule is triggered.	Planned (2)
The option to set nicknames for separate devices on the network.	Planned (2)
The option to set limits for monthly data usage per device on the network.	Planned (2)
The option to limit the amount of time (or times of day) usage per device on the network.	Planned (2)
Allow user to select from pre-compiled filter lists, such as ad block or other EasyList filter sets (https://easylist.to/)	Planned (2)
User presets where the administrator can select low, medium, or high security.	Planned (2)
Allow the user to filter well known protocols (HTTPS, HTTP, ICMP, FTP, etc.)	Planned (2)

A help wizard to assist users with configuring the device, especially in areas which may require technical knowledge.	Bells and Whistles (3)
A smart filtering mode to scan domains and IP addresses based on notoriety in an external database (VirusTotal).	Bells and Whistles (3)
Geolocation filtering to either allow or block traffic from certain regions.	Bells and Whistles (3)
A statistics page to allow users to view important system information, such as violations or user data.	Bells and Whistles (3)
A HTML page that tells the user when a site has been blocked	Bells and Whistles (3)

4. Software Engineering Tools and Techniques

The software development for the NanoFirewall device is best suited for a hybrid Agile-Waterfall approach. As a small project with a clearly defined firewall function, the traditional waterfall approach allows us to first design, then implement the core components, then test those components of our software. Once we have this core functionality written with a waterfall approach, we can transition to agile development to implement additional features. It makes sense for all the strategizing, planning, and core functionality to be squared away first before switching to Agile development since there are two major components of the project which need a well defined communication protocol between the two. With this approach, most of our system design is done up front. When we switch to agile development, sprints will last 5 days and be assigned during a team meeting on Fridays to be completed before another team meeting on Wednesdays. Team meetings and communication will be held on a Discord server with all team members.

Peer review will then be conducted during the time between Wednesday and Friday's team meetings, where changes will finally get merged. Peer review will be signed off by the corresponding partner for the module of the project the code is for. The peer reviewer should ensure that the code is well commented before checking in. The reviewer should also determine that the code has been reasonably tested. It is left to the developer whether a manual test or unit test is appropriate for a given feature.

All team members are expected to use GitLab for Git version control with descriptive Git commit messages, as well as the issue board for tasks and bug tracking. If a team member finds a bug, they are expected to create a new task for it with a bug label. If the issue is to create a new feature, they will instead use a feature label. The GitLab Wiki page will also store the documentation of the project and should be written "as-we-go" during development.

Web development will be done using the VS Code editor. C++ development will be performed using the Eclipse C/C++ Development Tools (CDT). The main libraries for our web interface will be Node.js and react.js. For our C++ code the libraries will be libtins for packet inspection, restbed for web communication, and the nlohmann JSON library.

5. Timeline

Week Number:	Evan Hirn Goals:	Tyler Klein Goals:	Hunter Moffat Goals:	Spencer Cameron Goals:
Phase 1: Alpha				
1: (1/19-1/24)	Config manager parsing MAC address and TCP/UDP port rules.	Create rule object for MAC address filtering and TCP/UDP port filtering.	Implement a login screen with actual authentication.	Implement login authentication. Verify authentication on backend API requests (in header)
2: (1/25-1/31)	Ability to remove filters from json config file.	Allow switching between block list and allow list.	Add configuration to switch between block list and allow list. Ability to remove filters.	Allow ability to remove filters using a web request.
3: (2/1-2/7)	Ability to save which devices the rules should apply to.	Handle rules that should only apply to certain devices.	Add device lists to rule pages (in addition to filter list). Add pages for MAC filtering and TCP/UDP port filtering.	Requests to handle adding certain devices to be filtered.
4: (2/8-2/14)	Framework testing	Framework testing	Framework testing	Framework testing
Phase 2: Beta				
5: (2/15-2/21)	Allow user to select from pre-compiled filter lists	Implement email reporting.	Create management page	Implement functionality to change admin email, username, and password.

6: (2/22 - 2/28)	Allow user to select from pre-compiled filter lists	Implement data limits for devices	Implement devices page with nicknames and limits.	Implement functionality to disable/enable/reboot firewall and change network settings.
7: (3/1 - 3/7)	Implement config manager presets	Implement time of day filtering	Implement protocol filtering page	Implement parser to parse data limits from json.
8: (3/8 - 3/14)	Implement config manager presets	Implement well known protocol filtering	Create a Preset page for users to select high/medium/low.	Implement nickname functionality.
Phase 3: Final Phase				
9: (3/15 - 3/21)	Optimizations to help with performance / multithreading. Race condition review.	Optimizations to help with performance / multithreading. Race condition review.	Add pages for smart filtering and geolocation filtering.	Implement parser to accept request to limit well known protocols.
10: (3/22 - 3/28)	Device Statistics	Smart Filtering (VirusTotal integration)	Implement statistics page.	Implement functionality to accept requests to apply a preset. Statistics requests
11: (3/29 - 4/4)	Blocked website HTTP replacement	Geolocation filtering	Make front end look pretty	Add request for smart filtering and geolocation filtering.
12: (4/5 - 4/11)	Integration testing.	Integration testing.	Integration testing	Integration testing.

6. References

[1] Adaware, *What is a firewall?* [Online]. Available: https://www.adaware.com/faq/firewall-frequently-asked-questions

[2] libtins, Introduction. [Online]. Available: http://libtins.github.io/#

[3] PiHole https://discourse.pi-hole.net/

7. UI/Use Case Appendix

Number	1
Title	(Hardware use case) Secure a home network
Preparer	Tyler Klein
Actor/User	Administrator of a local home or small-business network
User Story	As an administrator of a local area network, the user would want to increase network security by introducing a firewall into the system.
Course of Events	1. The user plugs an Ethernet cable between the WAN port on the wired version of the NanoFirewall and the LAN port on their router 2. The user plugs an Ethernet cable between the LAN port on the wired version of the NanoFirewall and whatever was regularly plugged into the router, such as an Ethernet switch. 3. The user powers the device, and now has network security with default settings.
Exceptions/Alternates	- The user may want an alternative hardware setup, such as Use Case 2 or 3. - The user may want to perform advanced configuration using the web interface.
Related UI	Sketch 1

Number	2
Title	(Hardware use case) Protect yourself on a wireless network
Preparer	Tyler Klein
Actor/User	A client of an unsecured public network
User Story	A client brings their laptop somewhere with free WiFi, such as an Internet cafe, coffee shop, or airport. The user wants more security and privacy from this network.
Course of Events	1. The user plugs an Ethernet cable between their laptop's Ethernet port and the wireless version of the NanoFirewall's Ethernet port. 2. The user configures the wireless NanoFirewall to connect to the unsecure public network. 3. The user now has added protections to the Internet from a public network.
Exceptions/Alternates	- The user may want an alternative hardware setup, such as Use Case 1 or 3. - The user may want to perform advanced configuration using the web interface.
Related UI	Sketch 2

Number	3
Title	(Hardware use case) Secure a wireless network
Preparer	Tyler Klein
Actor/User	Administrator of a wireless local home or small-business network
User Story	The same user story as Use Case #1, however with client devices being wireless rather than wired Ethernet.
Course of Events	1. The user plugs an Ethernet cable between the NanoFirewall and the LAN port on their router. 2. The user configures wireless access point settings using Use Case 17. 3. Devices on the network connect to the NanoFirewall's WiFi network 4. These wireless devices now have network security with default settings.
Exceptions/Alternates	- The user may want an alternative hardware setup, such as Use Case 1 or 2. - The user may want to perform advanced configuration using the web interface.
Related UI	Sketch 3

Number	4
Title	DNS Filtering
Preparer	Evan Hirn
Actor/User	Parent
User Story	As a parent, I do not want my child playing online browser games when he should be doing his homework. I would like to blacklist online web browser games.
Course of Events	1. Go to a device on the LAN that has a web browser. 2. Type in the firewall's local IP address into the URL bar.. 3. Login as administrator. 4. Click on the Rules button in the left panel. 5. Go to the DNS filter section 6. Create a new rule that blocks the sites my child should not be on 7. Click the add button
Exceptions/ Alternates	● The user could find a list of top browser game sites online and add it to their rule
Related UI	Sketch 5

Number	5
Title	Filter MAC addresses
Preparer	Tyler Klein
Actor/User	Configurer of the firewall
User Story	A user doesn't want new devices being plugged into their network, so it creates a block or allow list of MAC addresses that the firewall can process.
Course of Events	1. Use selects to change settings for MAC filtering on the left panel 2. User selects either block or allow list 3. User provides a list of MAC addresses 4. User selects block traffic or report to admin. 5. The firewall will then either block unallowed MAC addresses or report to the network admin if an unallowed MAC address is spotted.
Exceptions/Alternates	- The MAC address the user provides is malformed - The user tries to make both a block list and an allow list. This doesn't make sense.
Related UI	Sketch 9

Number	6
Title	Restricting child's access to websites
Preparer	Spencer Cameron
Actor/User	Parent configuring the firewall
User Story	A user wants to block their child's access to certain websites containing explicit content or certain web games.
Course of Events	1. User clicks on DNS Filtering on the left tab 2. User adds rule for the website to be blocked for their child 3. User clicks apply
Exceptions/Alternates	- The IP address/Domain is malformed - The selected user has admin privileges
Related UI	Sketch 5

Number	7
Title	Geolocation Blocking
Preparer	Tyler Klein
Actor/User	Configurer of the firewall
User Story	The user does not want foriegn traffic from countries where Internet privacy laws are not as strict, such as China and Russia.
Course of Events	1. User navigates to the rules dropdown 2. User selects the Smart Rules tab 3. User Enables Scan Domains/Scan IP boxes 4. User selects block country pull down 5. User checks the box on any relevant country they wish to block. 6. User clicks add. 7. The firewall will then check domains and IPs if they are from an unallowed country using an external API. The firewall may cache commonly accessed IPs to prevent slowing.
Exceptions/Alternates	- The user tries to make both a block list and an allow list. This doesn't make sense. - The external API has a limited number of lookups. - The external API cannot locate a given domain or IP.
Related UI	Sketch 10

Number	8
Title	Set child data limits per device.
Preparer	Spencer Cameron
Actor/User	Parent administrator of a local home network
User Story	The user doesn't want the people on their network (their children) to exceed a certain data limit per week/month.
Course of Events	1. User navigates to the connected devices page. 2. User clicks on the relevant connected device on this page. 3. User inputs the data allowance for the selected device.
Exceptions/Alternates	- Limit is invalid (negative number)
Related UI	Sketch 7

Number	9
Title	Virus Scanning
Preparer	Spencer Cameron
Actor/User	Administrator of a wireless local home or small-business network
User Story	The user wants to ensure that packets traveling into the network are safe and free from viruses.
Course of Events	1. User navigates to the rules section dropdown 2. User clicks on Smart rules 3. User enters VirusTotal API Key 4. User checks box, enable virus scanning 5. User clicks apply
Exceptions/Alternates	- The virus isn't found on virus total - File too large - Out of API lookups (limited)
Related UI	See Sketch 10

Number	10
Title	Security Presets
Preparer	Evan Hirn
Actor/User	Home firewall user
User Story	I just bought the firewall and I do not have the technical knowledge to set up custom rules. However, I want strong security for my home network. I want to change the default security preset from medium to high.
Course of Events	1. Go to a device on the LAN that has a web browser. 2. Type in the firewall's local IP address into the URL bar. 3. Login as administrator. 4. Click on the Rules button in the left panel. 5. Click on the security preset dropdown and change it from medium to high. 6. Click the add button.
Exceptions/ Alternates	
Related UI	Sketch 12

Number	11
Title	Creating custom rule (Not for average user)
Preparer	Spencer Cameron
Actor/User	Advanced administrator of a firewall who knows C++
User Story	An advanced user needs a rule that doesn't come pre-installed as part of our firewall.
Course of Events	1. User writes a C++ object that implements our defined abstract class/interface 2. User compiles their additional library into our firewall 3. User enables it in the rules tab.
Exceptions/Alternates	- Invalid code written
Related UI	No sketch applicable.

Number	12
Title	Toggling Rules
Preparer	Evan Hirn
Actor/User	Parent
User Story	As a parent, I like to limit my child's internet usage during the school year. I have previously made a rule that does so. I would like to toggle off this rule now that it is summer break.
Course of Events	1. Go to a device on the LAN that has a web browser. 2. Type in the firewall's local IP address into the URL bar 3. Login as administrator 4. User clicks on the rule they want to toggle 5. User unchecks "Enable" 6. Press the apply button
Exceptions/ Alternates	
Related UI	Any sketch for a rule. (ie Sketch 5)

Number	13
Title	Rebooting the Firewall
Preparer	Evan Hirn
Actor/User	Firewall administrator
User Story	My firewall recently started acting a little funny. I would like to reboot the firewall in order to fix this problem.
Course of Events	1. Go to a device on the LAN that has a web browser 2. Type in the firewall's local IP address into the URL bar. 3. Login as administrator 4. Click the management button on the left panel of the home page.. 5. Click the reboot button. 6. Wait 1 minute until the device has rebooted.
Exceptions/ Alternates	● If for some reason the user cannot get to the login web page. The user can unplug the device and then re-plug it in in order to restart
Related UI	Sketch 6

Number	14
Title	Disable firewall
Preparer	Spencer Cameron
Actor/User	Administrator of a wireless local home or small-business network
User Story	The user needs to turn off or temporarily disable the firewall.
Course of Events	1. User clicks on the management tab of the firewall UI. 2. User clicks disable firewall.
Exceptions/Alternates	- Firewall could be enabled as well
Related UI	Sketch 6

Number	15
Title	Changing Admin Password
Preparer	Spencer Cameron
Actor/User	Administrator of a wireless local home or small-business network
User Story	Admin needs to change the administrator password for the firewall.
Course of Events	1. User clicks on the management tab of the firewall UI. 2. Enters new admin password and presses apply
Exceptions/Alternates	
Related UI	Sketch 6

Number	16
Title	Network Settings (IP addresses or DHCP)
Preparer	Hunter Moffat
Actor/User	Business owner or parent in family (Network Administrator)
User Story	A business owner wants to set up their NanoFirewall to be wireless, requiring them to manually set up the IP address of the firewall.
Course of Events	1. The business owner logs into the NanoFirewall using administrator credentials. 2. The business owner navigates to the network settings tab. 3. The business owner sets the IP of the NanoFirewall to something they desire.
Exceptions/Alternates	- The wireless version needs wireless settings
Related UI	Sketch 6

Number	17
Title	Make web interface available only on LAN side, WAN side, or neither, or both
Preparer	Hunter Moffat
Actor/User	Network Administrator
User Story	A network administrator wants their wireless NanoFirewall to only be accessible from wired devices.
Course of Events	1. User logs in to the NanoFirewall using Administrator credentials. 2. User navigates to the "Management" tab. 3. Once in the management tab, the user selects eth0/eth1 boxes to enable/disable Lan/Wan.
Exceptions/Alternates	- Disabling both WAN and LAN will never be able to access the NanoFirewall again.
Related UI	Sketch 6

Number	18
Title	Time Restrictions
Preparer	Evan Hirn
Actor/User	Parent
User Story	As a parent, I do not want my child playing video games all night. Therefore I would like to set a restriction on what times in the day my child's devices are able to access the internet.
Course of Events	1. Go to a device on the LAN that has a web browser 2. Type in the firewall's local IP address into the URL bar. 3. Login as administrator 4. Go to the devices tab on the left side of the home screen 5. Select all of my child's devices 6. Create a new rule where those selected devices can only access the internet from 7am to 11pm 7. Press the apply button
Exceptions/ Alternates	- I could also make a rule where the devices only get 5 hours of internet access per day if I wanted to limit the screen time of my child
Related UI	Sketch 7

Number	19
Title	IP Filtering
Preparer	Hunter Moffat
Actor/User	Network Administrator / User
User Story	User is at a "sketchy" coffee shop using the public wifi to browse the web. They want to ensure that their device is safe from hackers also using this network.
Course of Events	1. User logs in to the NanoFirewall using Administrator credentials. 2. User Navigates to the rules dropdown tab and clicks on IP Filters. 3. Then the user specifies that they want to block all requests and responses from devices in the local host IP range. 4. Once the rule is created no devices connected to the network can send and receive data from their device.
Exceptions/Alternates	- IP is invalid
Related UI	Sketch 5 but with IP instead of DNS

Number	20
Title	Email Notifications
Preparer	Evan Hirn
Actor/User	Parent
User Story	As a parent, I do not want my young child watching explicit content. I want to be emailed every time my child goes to a well known explicit content website.
Course of Events	1. Go to a device on the LAN that has a web browser 2. Type in the firewall's local IP address into the URL bar. 3. Login as administrator 4. Go to the rules tab on the left side of the home page. 5. Go to the DNS blacklist setting. 6. Add explicit sites to the blacklist. 7. Check the option where the rule is only for specific devices, then add my child's devices. 8. Check the option where an email is sent when this rule is broken. 9. Go to the main settings and make sure my email address is set as the administrator email. 10. Press the apply button

Exceptions/ Alternates	• I could have made the rule global for all devices
Related UI	Sketch 5 and Sketch 6

Number	21
Title	Viewing Statistics
Preparer	Hunter Moffat
Actor/User	Business owner or parent in family
User Story	A business owner wants to see what websites their staff is using the most that is non work related. The business owner clicks the statistics tab on the NanoFirewall web page that displays the top domains accessed by their staff.
Course of Events	1. The business owner sees their staff spending too much time on various websites that are non work related. 2. The business owner goes to the statistics view of the NanoFirewall to see what non work related websites are being used the most. 3. The business owner then adds all of the most distracting websites to the NanoFirewall's block list. 4. The staff of the business owner can no longer access those websites and be distracted for work.
Exceptions/Alternates	
Related UI	See Sketch 4

Number	22
Title	Blocking unsecure HTTP traffic
Preparer	Tyler Klein
Actor/User	Configurer of the firewall
User Story	The network administrator only wants web browsing to occur over secure HTTPS. This means the firewall should block all HTTP traffic.
Course of Events	1. User navigates to the protocol filtering page 2. Use selects to block HTTP and allow HTTPS
Exceptions/Alternates	Traffic is neither http nor https
Related UI	Sketch 11

Number	23
Title	Port filtering
Preparer	Tyler Klein
Actor/User	Configurer of the firewall
User Story	The network administrator wants to block certain protocols by limiting which ports are allowed.
Course of Events	1. User selects to change settings for port filtering on the left tab in the Rules dropdown 2. User selects block list or allow list 3. User selects inbound or outbound port direction 4. User provides TCP ports to block/allow 5. User provides UDP ports to block/allow.
Exceptions/Alternates	- User provides an invalid port number - The user tries to make both a block list and an allow list. This doesn't make sense..
Related UI	Sketch 5 but with ports instead of DNS

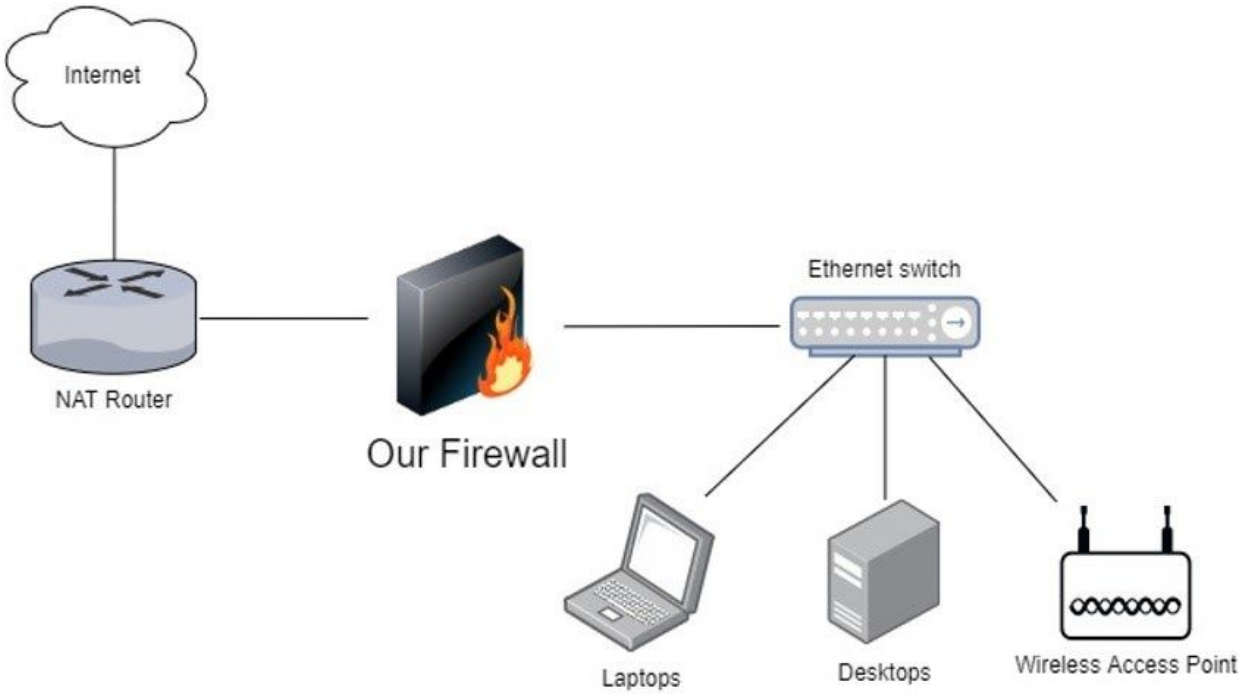
Number	24
Title	Home page
Preparer	Tyler Klein
Actor/User	Configurer of the firewall
User Story	The network administrator needs a landing point for the web interface, where it can then find the other pages of the management system.
Course of Events	1. The user goes to the IP address of the device in their browser 2. The user logs in 3. The user ends up on the firewall homepage
Exceptions/Alternates	
Related UI	Sketch 4

Number	25
Title	Adding a nickname
Preparer	Spencer Cameron
Actor/User	Firewall Administrator
User Story	The administrator may want to save someone's MAC address as a name so they can easily identify it for accessibility reasons.
Course of Events	1. User navigates to the devices panel 2. User clicks on the relevant device on this page 3. User clicks "Change nickname" 4. User enters relevant nickname
Exceptions/Alternates	- Nickname taken
Related UI	See Sketch 7

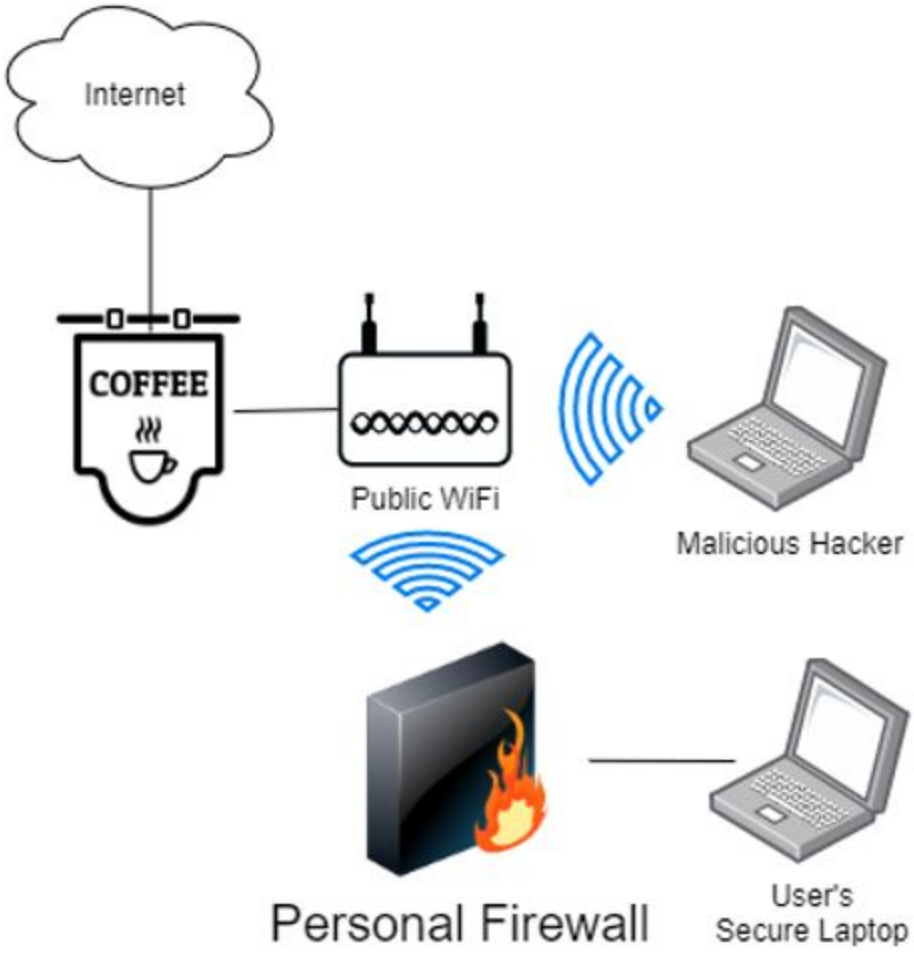
Number	26
Title	Login page
Preparer	Tyler Klein
Actor/User	Configurer of the firewall
User Story	The network administrator wants to manage the firewall, but needs to login to the device first
Course of Events	1. The user goes to the IP address of the device in their browser 2. The user is greeted with a login page 3. The user provides their username and password 4. The user is redirected to the home page
Exceptions/Alternates	- User gives an incorrect login - User forgets their password
Related UI	See Sketch 8

UI Sketches:

Sketch 1: home-network.jpg

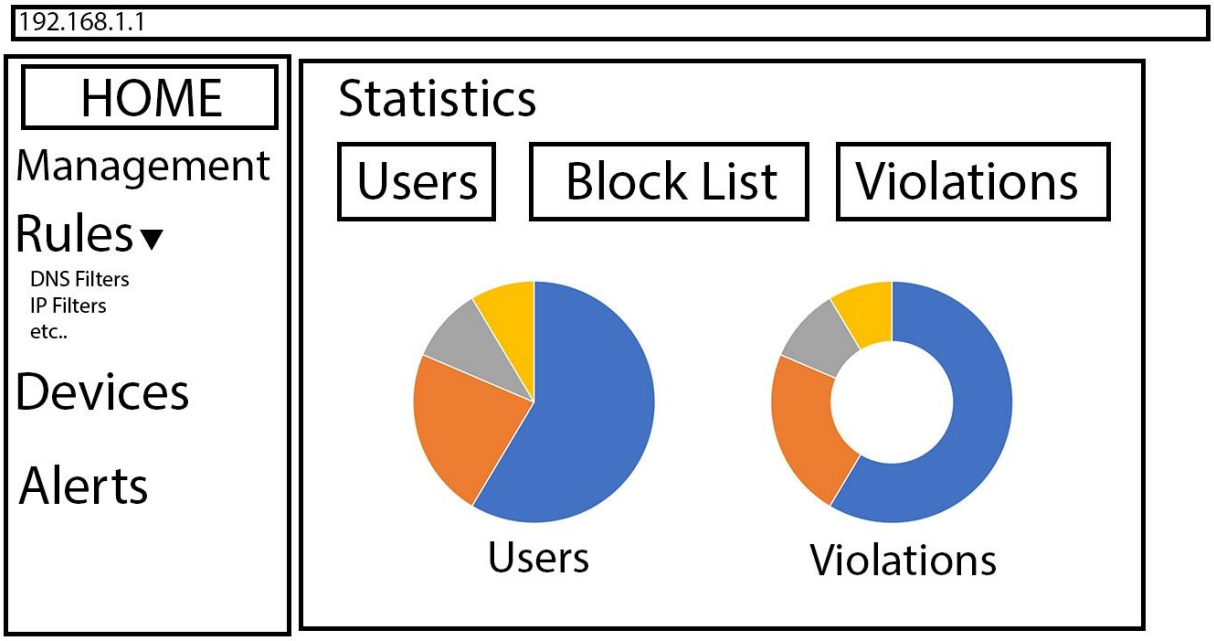


Sketch 2: public-network.jpg

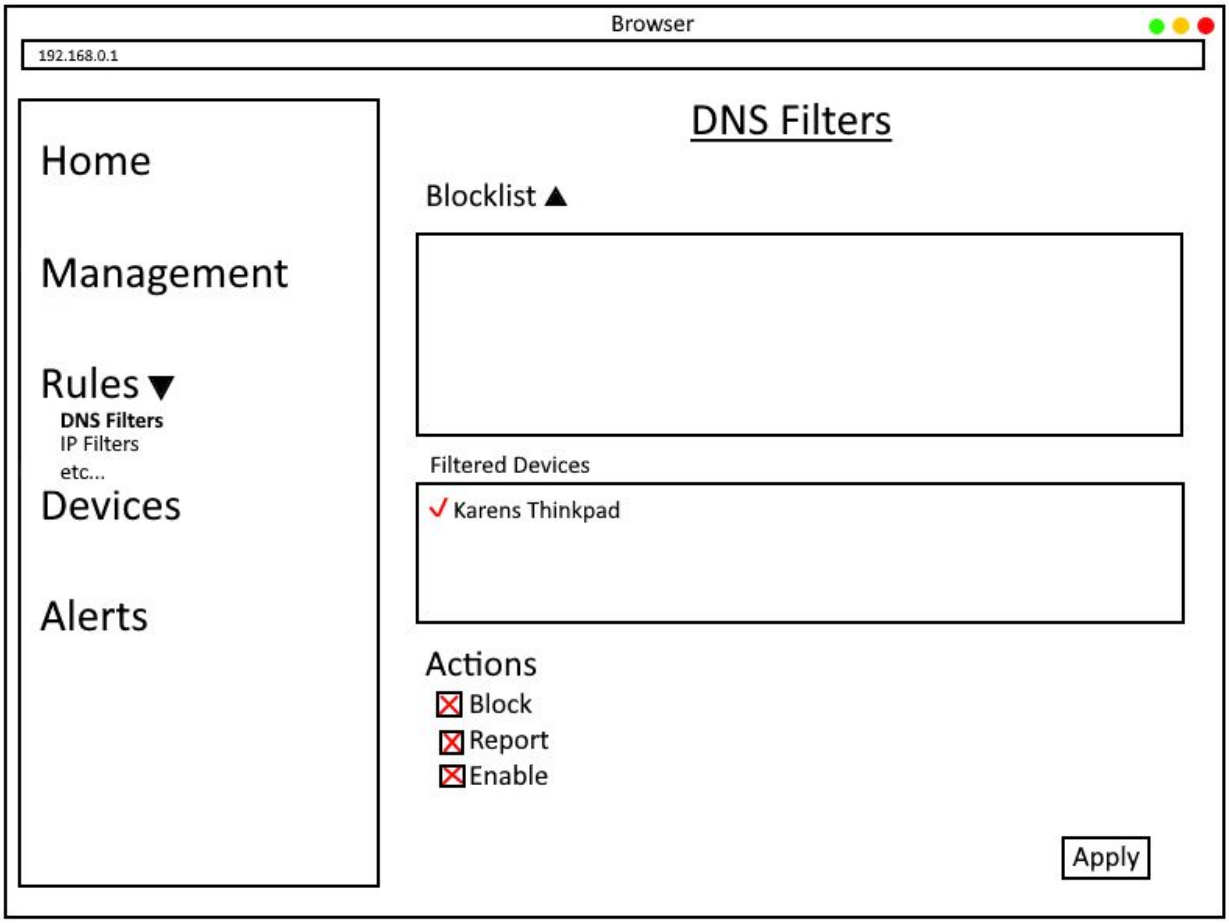


Sketch 3: wireless-network.jpg

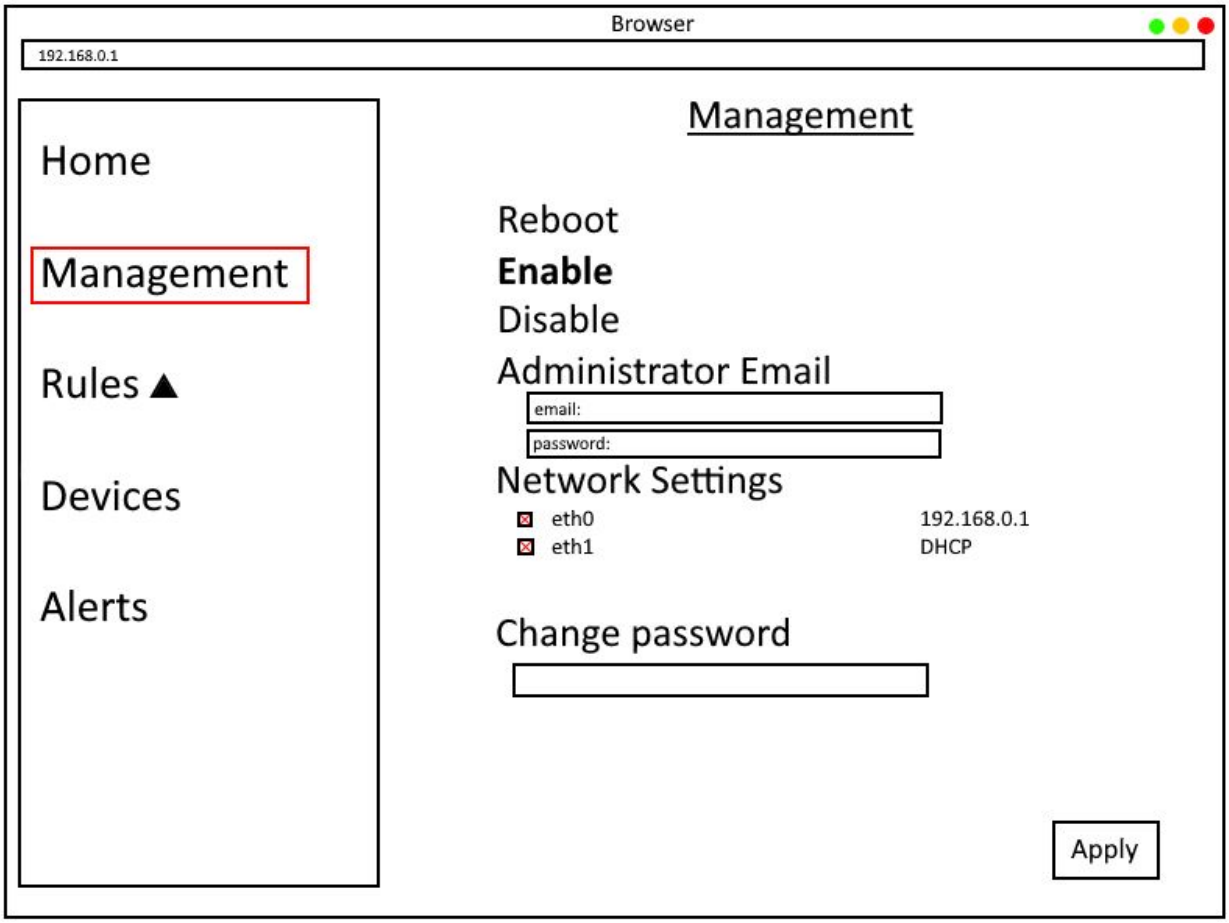
Sketch 4: home_page_ui.png



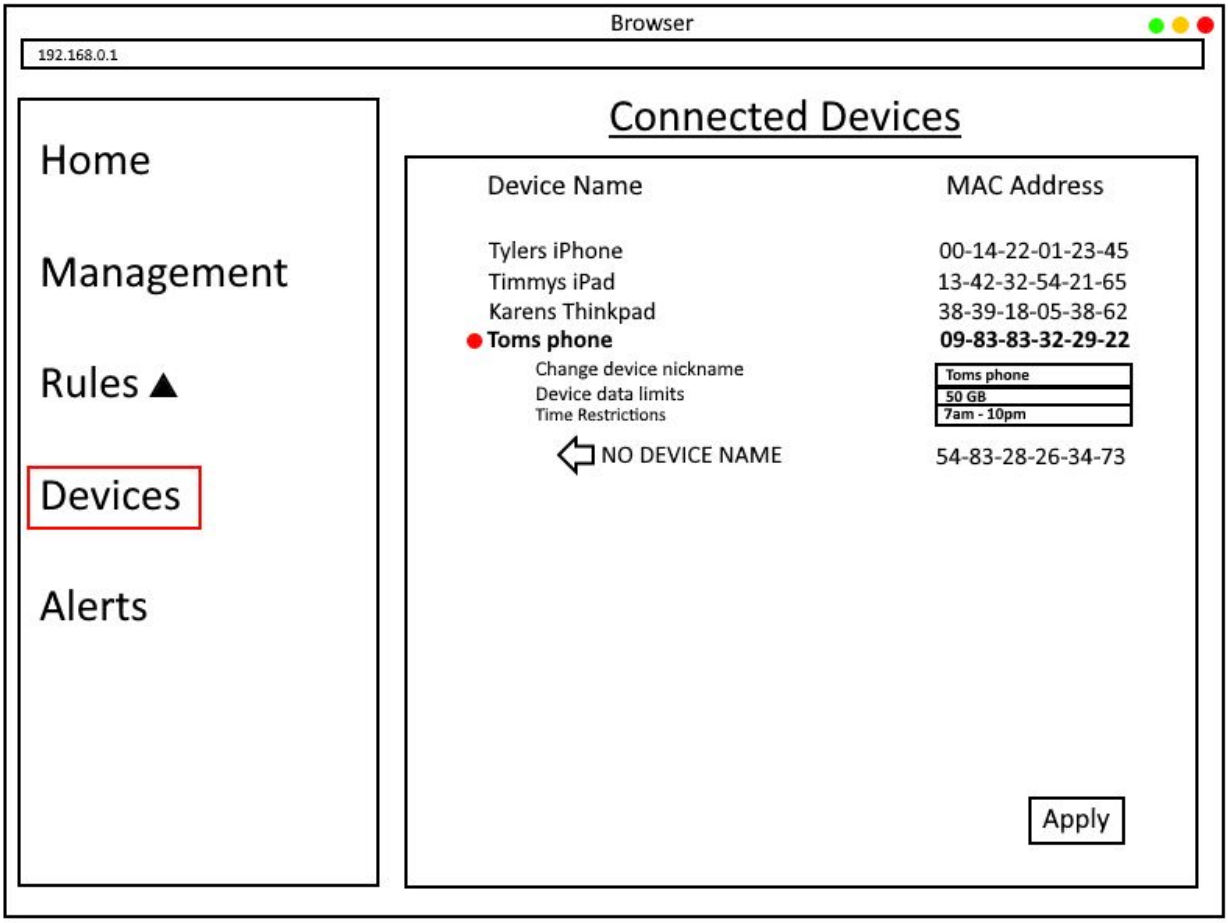
Sketch 5: DnsFilterPage.png



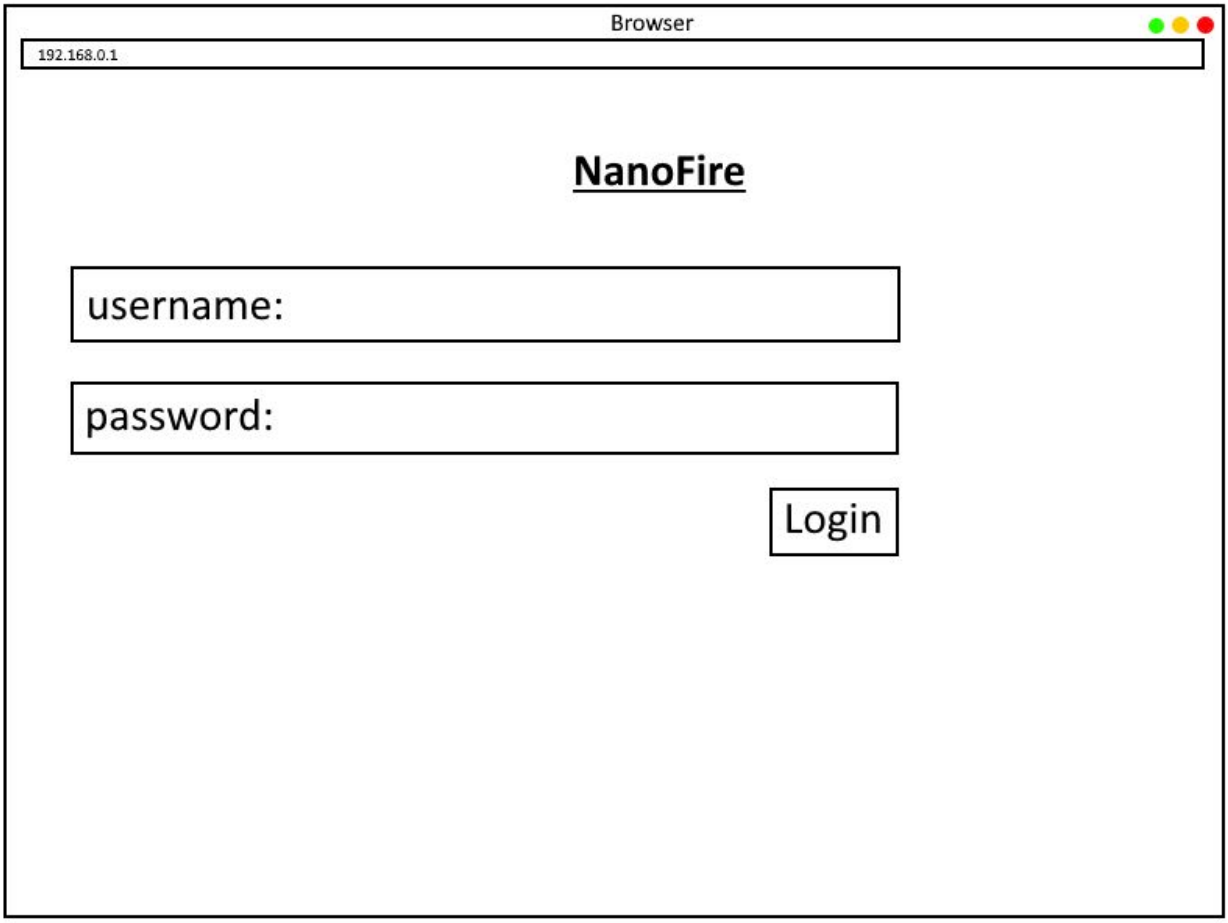
Sketch 6: ManagementPage.png



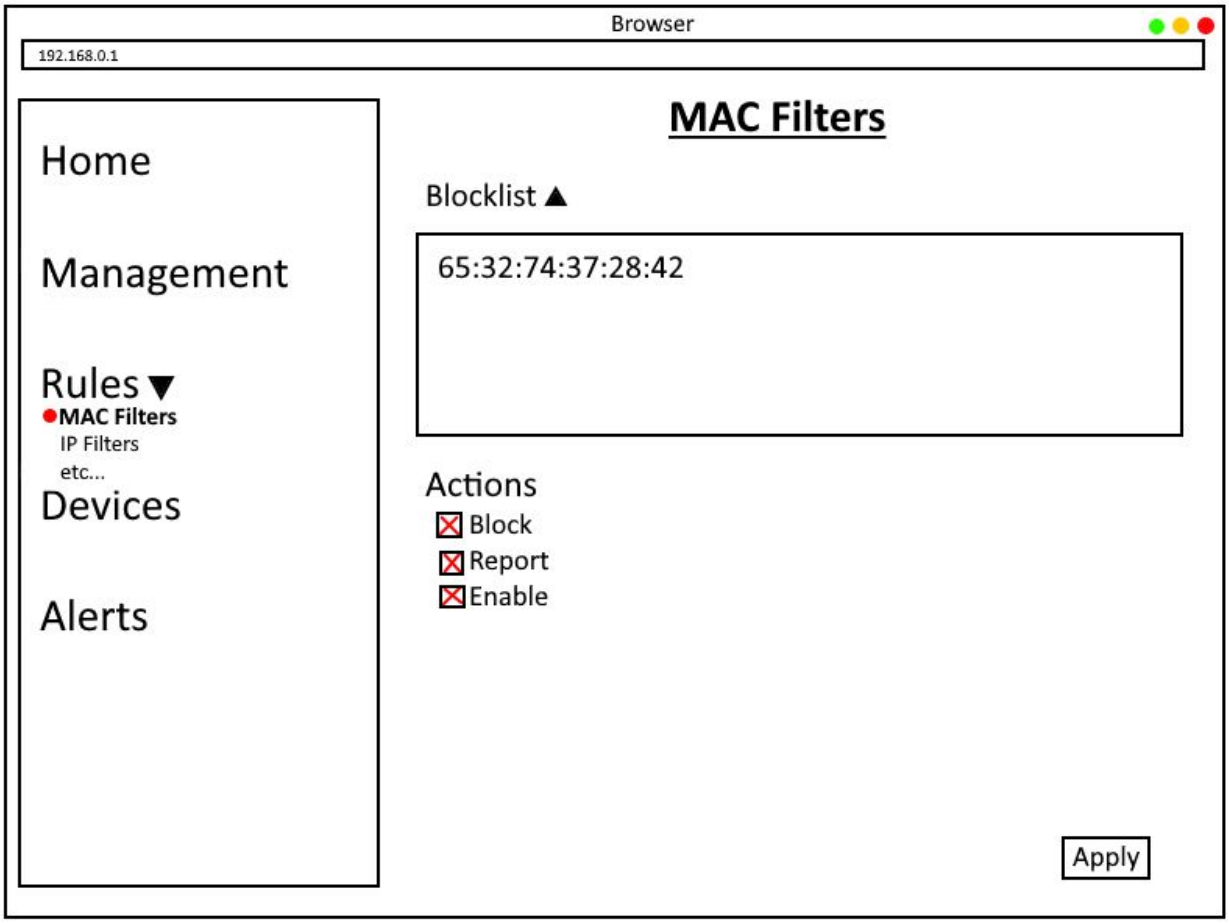
Sketch 7: DevicePage.png



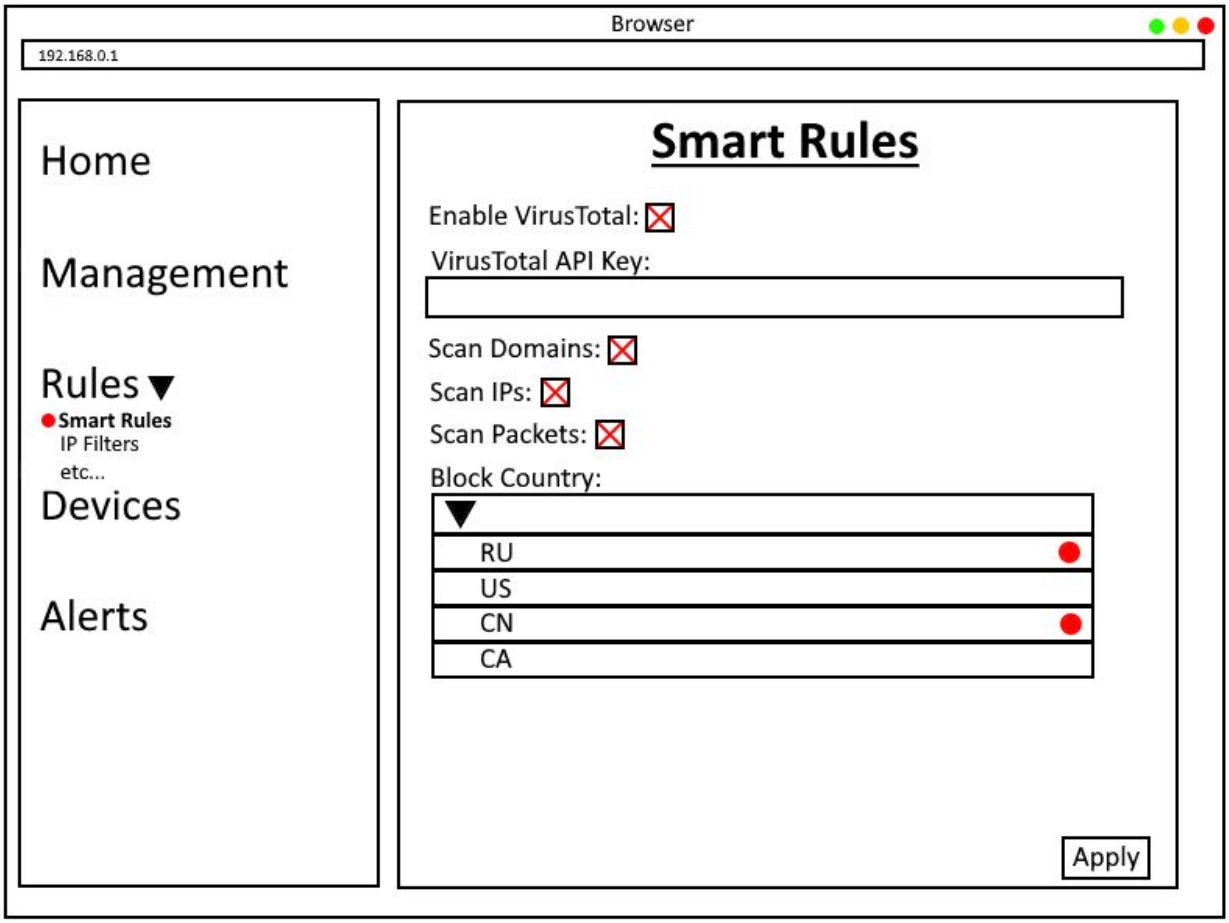
Sketch 8: LoginPage.png



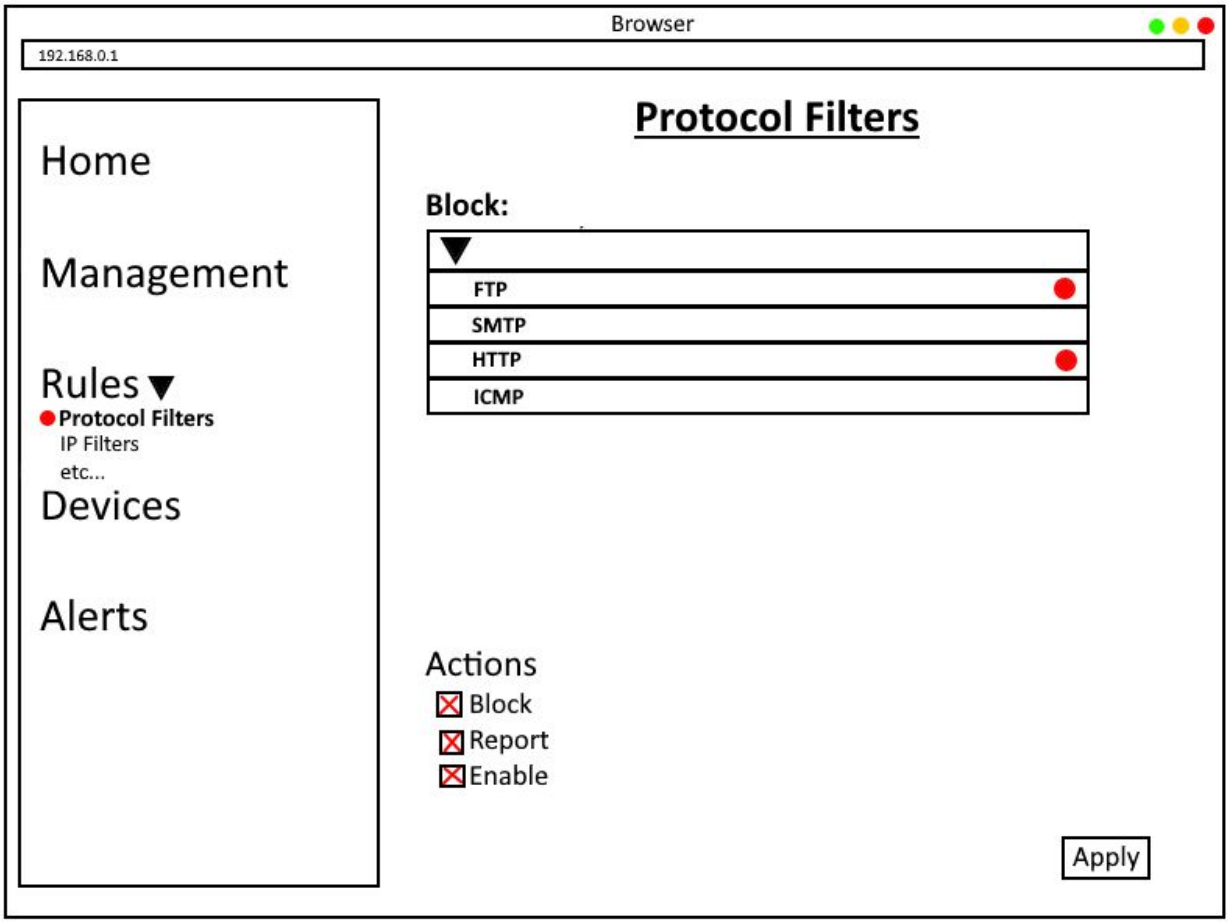
Sketch 9: MACFilterPage.png



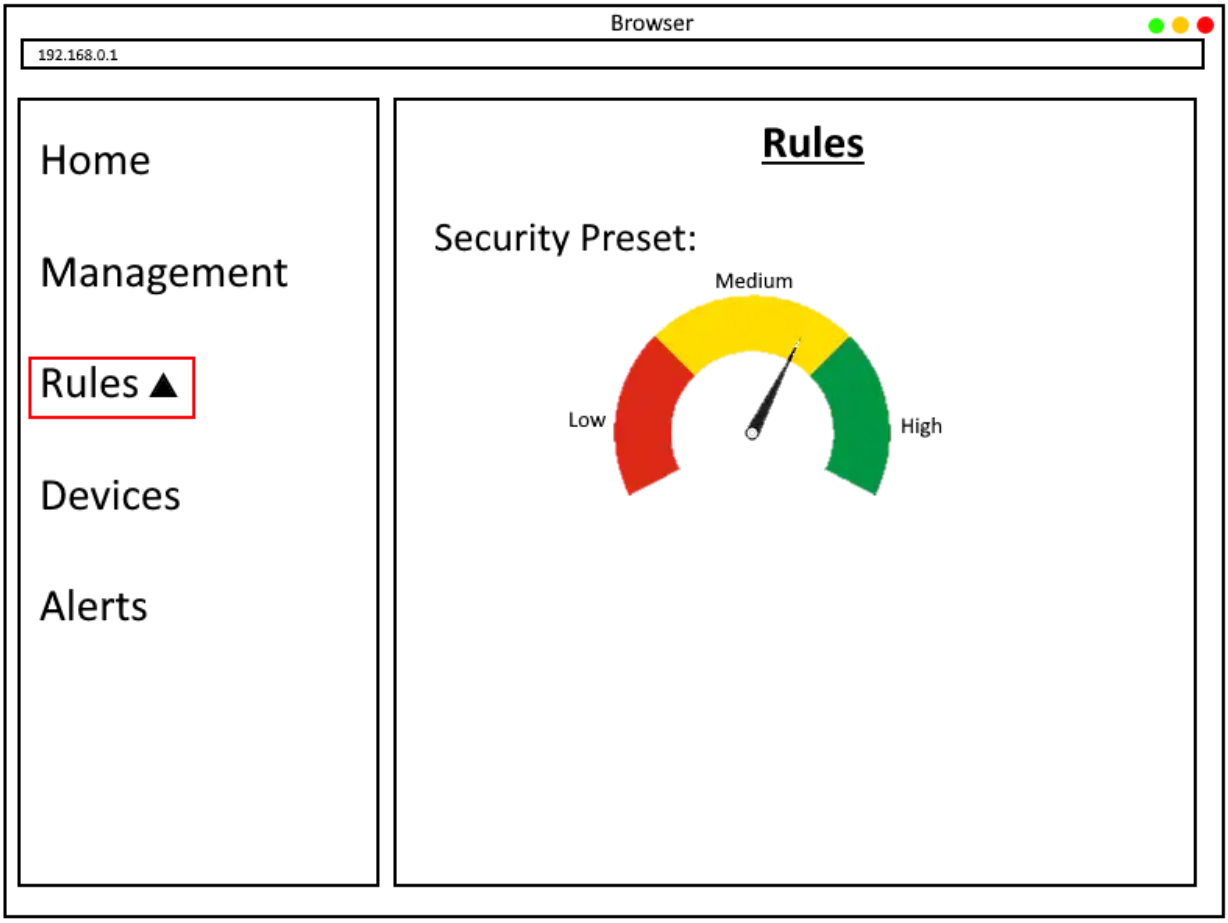
Sketch 10: SmartRulesPage.png



Sketch 11: ProtocolFilterPage.png



Sketch 12: RulesPage.png



8. Revisions Appendix

Section 1	Minor grammatical changes
Section 2	Updated required technology
Section 3	There were considerable changes to our architecture that needed to be accounted for. Most significantly, our web backend was originally a separate process but instead was combined into our C++ program. This cleaned up our architecture quite a bit. We also used bullet points when rewriting this section, as suggested by Ambuj.
Section 4	No Changes
Section 5	New section
Use Case Appendix	Updated numbering so that there are no skipped numbers.