International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 7, No.5 (Sep-2018)
http://dx.doi.org/10.12785/ijcds/070501

Enterprise-level Hardening of Web Browsers for Microsoft Windows

Ananth A. Jillepalli[1,2], Daniel Conte de Leon[1,2], Frederick T. Sheldon[2] and Michael A. Haney[1,2]

[1] *Center for Secure and Dependable Systems*
[2]*Department of Computer Science*
University of Idaho, Moscow, Idaho, United States

Received 23 May 2018, Revised 23 Jun. 2018, Accepted 12 Jul. 2018, Published 1 Sep. 2018

Abstract: Today, web browsers are a major avenue for cyber-compromise and data breaches. Web browser hardening, through high-granularity and tailored configurations, can help prevent or mitigate many of these attack avenues. For example, an enforced configuration that allows users to use one browser to connect to critical and trusted websites and a different browser for untrusted websites, with the former web browser restricted to trusted sites and the latter with JavaScript and Plugins disabled by default, can help prevent JavaScript- and Plugin-based attacks. However, most organizations today, still allow web browsers to run with their default configurations and allow users to use the same web browser to connect to trusted and untrusted websites alike. In this tutorial article, we describe in detail the steps needed for hardening the enterprise browser ecosystem using such tailored and high-granularity hardening approach at the enterprise scale by using the Windows Group Policy Editor and Active Directory Services, which are in widespread use in most organizations. We hope that system administrators use this guide to jump-start an enterprise-wide strategy for implementing high-granularity application-level hardening. This will help secure enterprise systems at the client-side, in addition to the network perimeter and server-side.

Keywords: Application hardening, Application-level least privilege, Phishing prevention, Secure configurations, Security policy, Web browser security

1. INTRODUCTION

Today, due to the flexibility and economic advantages offered by web technologies, organizations are moving legacy information technology systems, and in some cases operational technology systems toward web technologies at a rapid pace. Most cloud-based services also use web browsers for client and administrative access. This wide and rapid migration has resulted in web browsers being used today for accessing critical and private enterprise data and systems. This, while at the same time, synchronously or asynchronously, the same web browser is being used to access untrusted sites and browse the Web at large.

Modern web browsers implement full virtual execution with respect to Turing completeness [1]. This functionality is one of the major reasons for their success. However, it also allows attackers to remotely execute code of their choice by simply motivating the user to perform one click. In other words, under the current usage and configuration scenario, one click is the only-thing that separates a trusted and critical application from the rest of the untrusted Web. Browsing ecosystems configured in this manner, or not configured at all, violate at least two of Saltzer & Schroeder's [2] secure design principles: Least Privilege and Fail-safe Defaults.

A. The Problem and Threat Model

The problem results from the combination of: (a) the widespread and shared usage of web-browsers, (b) their compute, storage, and networking functionality, and (c) their default permissive security configurations. This creates a very vulnerable browsing ecosystem. Under these usage and configuration conditions, attackers have direct and remote access to the same client web browser that is used to access critical enterprise sites. This web browser is usually the user's, and sometimes system administrator's preferred web browser. Attacks in such environment can be accomplished by malicious actors through a simple web drive-by, phishing, or any other uniform resource locator (URL) sharing method that can lead users to a malicious website.

E-mail address: {ajillepalli, dcontedeleon, sheldon}@ieee.org, mhaney@uidaho.edu

Under these current usage and configuration patterns, it should not be surprising that many of today's attacks begin with the Web Browser. Also, it should not be surprising that attacks through the Web Browser are very common and successful at high rates. According to Verizon's 2016 Data Breach Investigations Report [3], the *Web App Attacks* pattern was used in 40% of the reported data breaches in 2015 (908 breaches with n=2,260); Though, the same pattern accounted for less than 10% of the reported incidents (5,334 incidents with n=64,199) [3]. For incidents and breaches in 2016, according to Verizon's 2017 Data Breach Investigations Report [4], the *Web App Attacks* pattern was used in about 30% of the reported data breaches (571 breaches with n=1,935); Though, the same pattern accounted for about 15% of the reported incidents (6,502 incidents with n=42,068) [3] p. 38.

This data indicates that the *Web App Attacks* pattern, during the last two years, has resulted in a high likeliness of breach. To further back up these assertions, we point out that in a presentation given at the USENIX Enigma 2016 Conference, Mr. Rob Joyce, Chief of the Tailored Access Operations Office at the U.S. National Security Agency, pointed out that today, most intrusions are carried out through one of these three initial vectors: 1) email (including phishing), 2) malicious website, and 3) malicious removable media [5].

Current security practices such as edge firewalls, network security perimeters, and network segmentation are not enough to adequately combat these Trojan horse-like attack avenues. Even the highest levels of perimeter security and server-side hardening cannot adequately protect against phishing attacks, when the same web browser is being used to access critical services and browsing the Web at-large. The inadequacy is especially true when the same set of permissive security configurations are used for all sites, trusted and untrusted.

As a result, the web browser is, unarguably, the weakest link on the enterprise today. However, most well-known and commonly used desktop web browser applications are designed and developed by teams with cybersecurity expertise and using secure development best-practices. These applications are also updated on a continuous basis. Though, the browser applications themselves may have a few vulnerabilities, the major problem currently resides not on application binary code but on application configuration. Most organizations today use a one-size-fits-all approach to application configuration. Also, most organizations allow browsers to be run with their permissive default configurations, which were not created for enterprise-level security. To solve this problem, we need an approach in which organizations tailor configurations of web browsers, and other applications, in a way that implements the principle of least privilege, to its maximum extent possible.

B. *The Contributions of This Article*

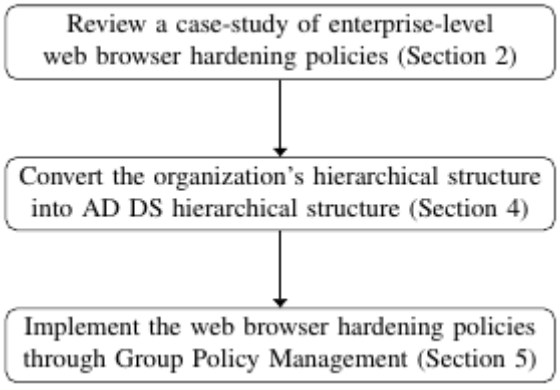

Figure 1.A flowchart of this article's contributions. AD DS = Active directory Domain Services.

The problem described in the previous section needs to be addressed today. To that end, we describe a step-by-step walkthrough that can be used, today, by system administrators to harden the web browser client infrastructure in the enterprise.

The security policy used as a case study in this article enforces least privilege for Internet versus Intranet web applications. In this policy, two browsers, Internet Explorer and Microsoft Edge will be configured to enable all browser functionality, but for trusted Intranet sites only. A second browser, Google Chrome, will be configured with high security configurations to enable safer browsing of the Web at large, without having to continuously update and manage a website white-list. In this article, all steps needed to implement this browser hardening policy are described for an organization that uses Microsoft's ADDS and Group Policy. Figure 1 represents a flowchart of this article's contributions. We do not discuss about Mozilla Firefox in this article because Firefox has no native support for Group Policy Object (GPO) configurations.

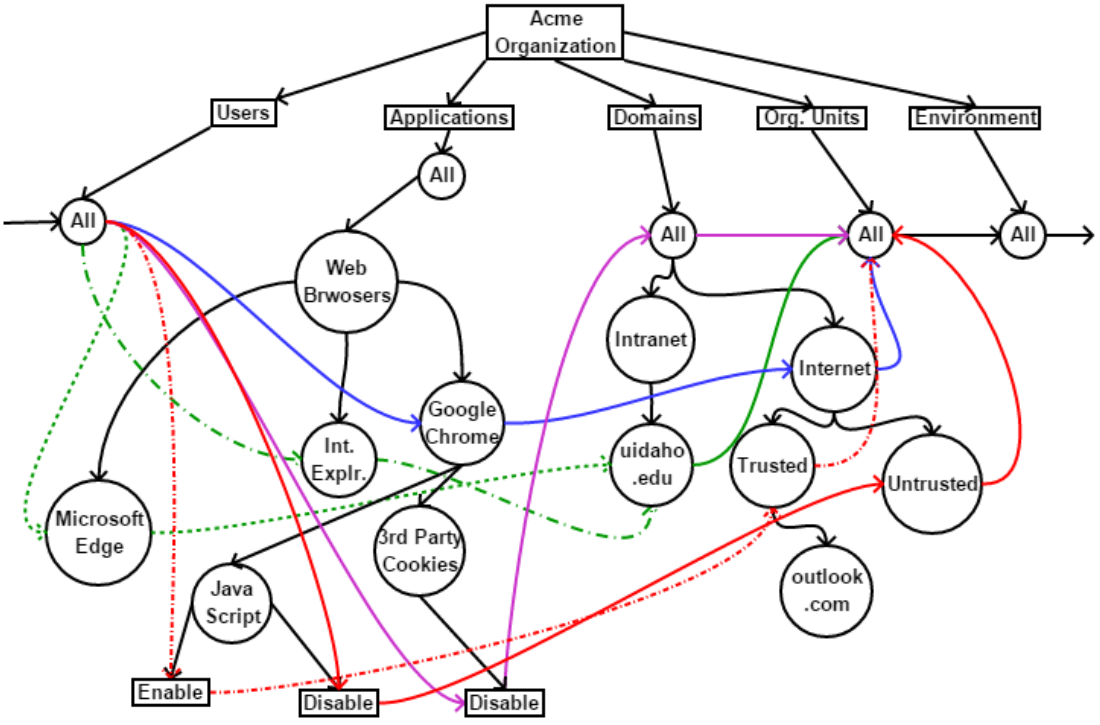

Figure 2. Pictorial representation of the hardening policy.

This article is an expanded version of a published conference paper [6]. The additional contributions of this article are as follows: 1) added a case study for describing a set of web browser hardening policies (Section 2); 2) added Microsoft Edge as one of the web browsers being configured (Sub-Section 5-B3); 3) expanded the background section by expanding explanations of Active Directory Domain Services, Group Policy Objects and Management, and Environment Setup (Sub-Sections 3-A, 3-B, & 3-C); 4) added the related work section (Section 6); 5) added new figures and flowcharts to better explain the contributions of this article (Figures 1, 2, 3, 6, 8, & 10); 6) better clarified the process of converting an organizational structure into an ADDS structure (Section 4 and Figures 4 & 5).

C. The Outline of This Article

The rest of this article is organized as follows. In Section 2 we describe the hardening policy target of this tutorial. In Section 3 we introduce the terminology and tools needed for remote application configuration in Windows. In Section 4 we describe the process of implementing an organization's hierarchical structure into an Active Directory Domain Services (ADDS) hierarchy. In Section 5 we detail all the steps needed to implement the proposed web browser hardening policy. In Section 6 we briefly describe related research work. We present our conclusion in Section 8. A complete list of abbreviations, acknowledgments, and references is provided at the end.

2. WEB BROWSERS HARDENING POLICY CASE STUDY

The web browser hardening policy that we use as a case study in this article is described here. Other, multiple-application policies may be deployed using the steps described in this article.

1. Internet Explorer and Microsoft Edge shall be used only for connecting to the `www.uidaho.edu` Intranet web domain. This policy is represented with green-colored lines in the Figure 2.
2. Google Chrome shall be used only for connecting to websites other than Intranet domains. This policy is represented with blue-colored lines in the Figure 2.
3. JavaScript functionality in Google Chrome should only be allowed for trusted websites on the Internet. For example: `www.outlook.com`. This policy is represented with red-colored lines in Figure 2.
4. Third-party cookie functionality should be disabled in Google Chrome or Microsoft Edge for all untrusted sites. This policy is represented with purple-colored lines in the Figure 2.

Currently, there are no configuration options for Internet Explorer, Microsoft Edge, and Google Chrome that would block access to a list of websites (as of 05/20/2018). As such, in this article, we describe how to block JavaScript and Plugin functionality for untrusted sites. In the future, we intend to expand this work to include firewall configurations.

3. BACKGROUND

In this section, we introduce the services, concepts, and nomenclature needed to understand Microsoft's Active Directory Domain Services (ADDS) and be able to remotely manage Windows-based users and computers in an enterprise. This knowledge is necessary to design and remotely implement configurations on Web Browsers within managed Windows clients.

A. Active Directory Domain Services

Active Directory (AD) is Microsoft's brand name for a suite of remote administration and configuration tools. AD includes directory, identity, and remote configuration services. A Windows Server that is used to remotely control and configure a group of Microsoft Windows clients is called a *Domain Controller*. When AD services are added, as a server role, the term *Active Directory Domain Services* is used (ADDS) [7]. ADDS enables administrators to organize enterprise assets and configurations using a five-level hierarchy. The root of the structure is called an AD Forest because multiple AD Servers act as peers for fault tolerance. At the second level are Domain Controllers (Windows ADDS Servers). Multiple Domain Controllers can manage and replicate the organization's directory(ies) and user and device configurations, in whole or in part, and in a distributed and fault-tolerant fashion. Within this level, we have *domains*, which are the major grouping entity within the ADDS model. At the third level are Organizational Units (OUs). At the fourth-level are Users and Computers which are attached to OUs, these are the managed assets. At the fifth level are configuration options and their respective values, which in ADDS are called *Group Policy Objects* (GPOs). These are applied to whole OUs using a sequential priority. This way, a given set of configurations, the GPO Objects, can be remotely enforced in all Users and Computers within an Organizational Unit. In ADDS the height of the hierarchy is fixed.

B. Group Policy Objects and Management

Group Policy Objects (GPOs) is a term to define sets of configuration settings and their corresponding configuration values. In the ADDS model, configurations and their values must be copied within each OU to be applied to the OU or they can be applied to the whole domain instead, by attaching all the OUs to a domain and creating a GPO list linked to the domain (more information on this process in Subsection 4-C). The *Group Policy Management Console* [8] is the tool used to create and edit GPOs based on available configuration templates called Administrative Templates. These templates are stored in ADMX/ADML [9] files.

In Windows clients, usually at client boot time, the server side ADDS software communicates with the client-side Group Policy software through the IP Network. Up to 20 different ports and 40 different network protocols are used for all ADDS services and about 5 of each for Group Policy [10]. Then the client-side software uses the received Group Policy information to populate Windows Registry keys and their corresponding values. This is what makes the configurations effective. Only systems and applications that support configuration through Microsoft's Group Policy, and that use the Windows Registry for application configuration, can be configured this way. Internet Explorer has extensive support for GPO. Google Chrome has very good support. Microsoft Edge currently supports a limited set of configuration options, though support for additional configuration options is expected to increase. Mozilla Firefox has no native support for GPO configurations nor it uses the Windows Registry for most local configurations, instead it uses a local configuration file. There exists a third-party add-on, known as 'GPO For Firefox' [11], to enable Firefox's configuration via group policy. However, the add-on is outdated and incompatible with latest version of Firefox (as of 05/20/2018).

C. Environment Setup

For the purposes of this tutorial we assume that the reader has access to an ADDS testing infrastructure that includes at least one Microsoft Active Directory Domain Services (ADDS) Server and at least two managed Windows clients. In a previously published open-access cyber-security tutorial [12], we describe a step-by-step guide for setting up an ADDS server, and the process of assigning a domain controller to a domain network. We direct the readers to read through this tutorial if they would like guidance on how to setup an ADDS network and assigning a domain controller to the network.

The activities described in this tutorial were performed on a single workstation, using three virtual machines (VMs): One Windows Server 2016 Datacenter 64-bit Build 14393 with ADDS role VM; and Two Windows 10 Education edition VMs, referred as *Clients* or *Client1* and *Client2* and attached to the ADDS Server.

4. CREATING AN ADDS STRUCTURE FOR THE ORGANIZATION

In this section we describe how to map the structure of an organization into an ADDS tree, that can be used to remotely apply configurations to a selected group of users or computers contained within an organizational

unit (OU). Figure 3 represents a flowchart of the steps involved in this section.

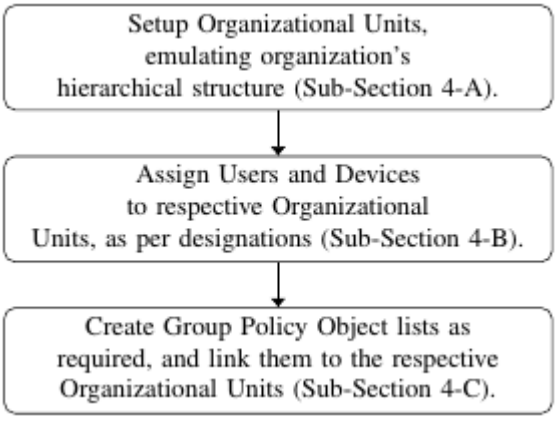

Figure 3. A flowchart representing steps involved in conversion of an organizational structure into an AD DS structure.

In ADDS an organization's departmental layout must be translated into groups called `Organizational Units` (OUs). However, in ADDS OUs cannot be nested. In other words, an OU cannot be the parent of another OU. If an organization has a deeper hierarchical structure such as *Campus: College: Department: ...* these sub-structures must be flattened. The lowest or leaf-level in ADDS is users or computers, both which can be attached to one and only one organizational unit (OU).

To summarize the process, the following four steps are required for the translation: (1) Create OUs as necessary, (2) Attach users/computers to respective OU, (3) Create GPO lists, and (4) Link respective GPO list to the relevant OU, as per organizational policy requirement. Figures 4 and 5 represent an example organizational structure and a corresponding ADDS structure. Figure 6 represents Figure 5 in the GUI of Group Policy Management interface. Next, we describe how to create an OU in ADDS.

A. Setting up Organizational Units

Here we detail the steps needed to create an organizational unit in ADDS.

1. On the Server, open `Server Manager` suite, if it is not automatically opened by default on startup.
2. Click on the `Tools` menu, in the top-right part of `Server Manager` suite's `Dashboard` screen.
3. In the `Tools` drop-down menu, click on the `Group Policy Management` selection. This action should produce the `Group Policy` Management interface box.
4. In left pane of the `Group Policy Management` interface box, identify the target forest and subsequently, the target domain.
5. Right-click on the target domain name and in the pop-up menu, select `New Organizational Unit`. In the subsequent dialogue-box, type-in the intended name of the OU to be created.
6. As a result, the OU is created. To verify this creation, one can find the new OU within the domain tree (Figure 7).

B. Attaching Users and Computers to OUs

Once an OU is created, objects (groups of users and/or computers) can be attached to the organizational unit. Here, we detail the steps to attach objects (users or computers) to an OU.

1. On the Server, open `Server Manager`, if it is not automatically opened by default on startup.
2. Click on the `Tools` menu, in the top-right part of `Server Manager` suite's `Dashboard` screen.

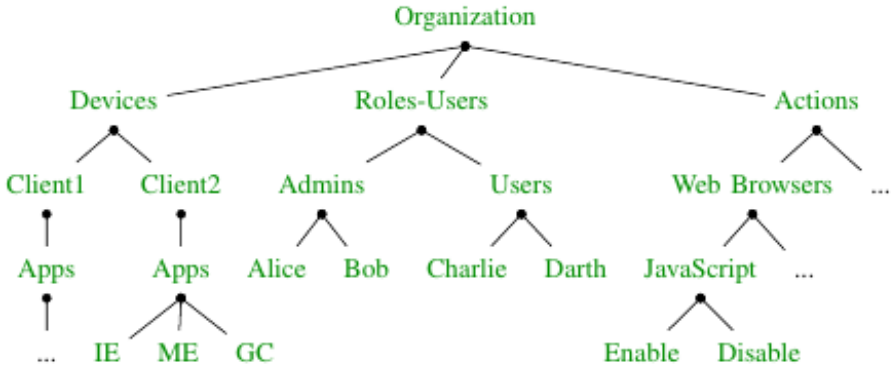

Figure 4. Example organizational hierarchy showing Devices, Roles/Users, Applications, and Actions (Permissions). IE = Internet Explorer, ME = Microsoft Edge, and GC = Google Chrome.

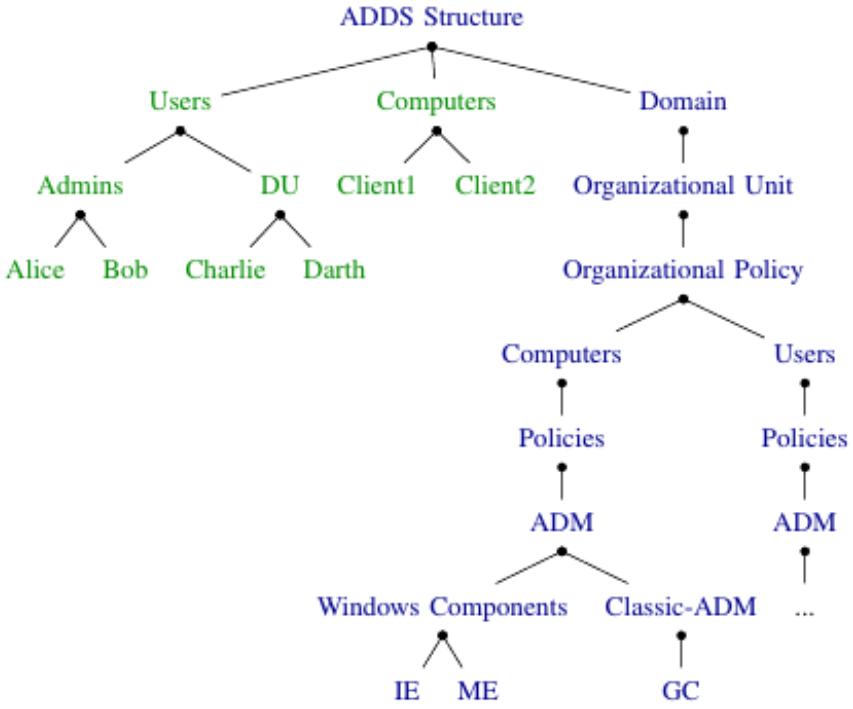

Figure 5. The ADDS hierarchy corresponding to the example organizational structure shown in Figure 4. Green-colored text is a direct mapping of entities from Figure 4. DU = Domain Users, ADM = Administrative Template.

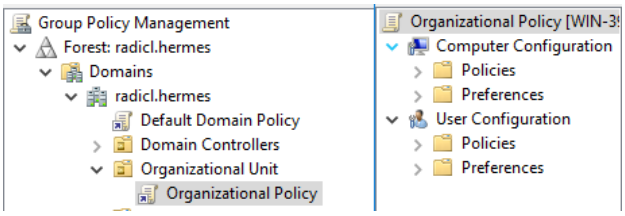

Figure 6.The resulting ADDS GUI showing the hierarchy model shown in Figure 5.

3. In the `Tools` drop-down menu, click on the `Active Directory Users and Computers` selection. This action should produce `Active Directory Users and Computers` interface box.
4. In this interface box, search for the target domain name tree, on the left pane of the interface box. Once identified, expand the target domain name tree.
5. Under the target domain name tree, identify the target object, which would be stored in its' respective folder. For example, if the target object is a user named `Joe`, it can be found in the `Users` folder.
6. Once the target object has been identified, right-click on the object to activate a menu pop-up. In this menu, select the item `Move...`. In the subsequent `Move` popup box, select the desired
7. OU as the destination to move the object to, then click `OK`.
8. The target object has now been successfully attached to the desired OU. To verify, in the `Active Directory Users and Computers` interface box, search for the target domain name tree, on the left pane of the interface box. Once identified, expand the target domain name tree.
9. Under the target domain name tree, search for the desired OU folder. Click on it. Press the `F5` key on the keyboard or click on the `Refresh` icon in the top tool-bar. If the target object is visible in the center pane of the desired OU, we can consider the attachment to be verified.

C. Creating a GPO and linking it to a desired OU

Once an Organizational Unit (OU) is created, and once all the desired individual users or groups of users and computers have been attached to the OU, we can now create and apply a set of configurations to all entities within the OU. To do this, we must first create a Group Policy Object (GPO) and link it to the target OU. Here, we detail the steps to create a GPO and link it to the desired OU.

1. In the `Group Policy` Management interface box, identify the target OU. Once found, right-click on it.

2. In the resultant pop-up menu, click on the `Create a GPO in this domain and Link it here...` selection.

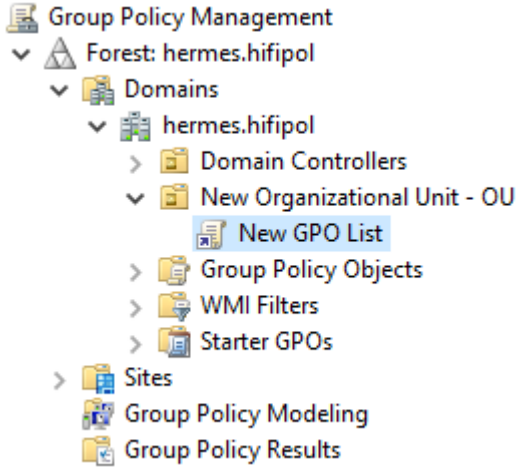

Figure 7. An example of (New Organizational Unit (OU)) and Group Policy Object list (New GPO List).

3. In the `NEW GPO` pop-up box, enter the desired name for the GPO list to be created and click on the `OK` button. Previously existing GPO lists can be inherited into a newly created list, if the old GPO list's configurations are supported the new GPO configurations. If one already has a GPO list and would like to reuse it, before clicking the `OK` button, one may want to select the `Source Starter GPO` button, to inherit the existing list into the new list.
4. Once a GPO List is successfully created and linked; it will be shown under the designated entity, which can either an OU (as shown in Figure 7) or a domain itself.

5. Hardening Policy Implementation Steps

Once the organizational hierarchy is translated into ADDS, as discussed in Section 4, we can edit the GPO list settings for the respective OUs to change configuration of domain system applications. In this section, we will go through the process of remotely hardening web browsers (more information on web browser selection in SubSection 3-B), by changing group policy settings.

This section also explains how `Administrative Templates` (ADML/ADMX/ADM) files can be imported into the `Group Policy Management Console`. Group Policies for Google Chrome and Microsoft Edge are not available on vanilla installations of Windows Server 2016. We need to download the corresponding administrative template packages and import them to incorporate Google Chrome and Microsoft Edge Group Policies. This section also describes how to accomplish such an import action. Figure 8 represents a flowchart of the steps involved in this section.

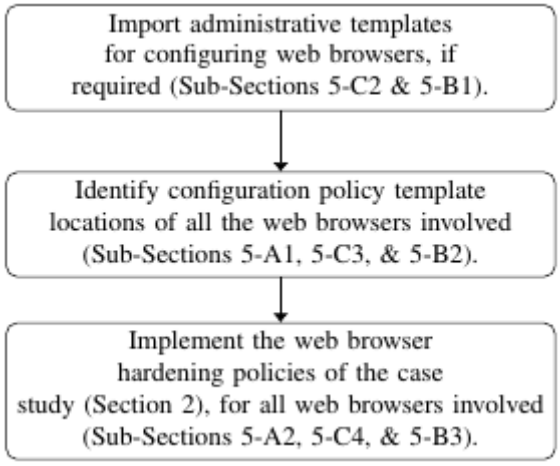

Figure 9. A flowchart representing steps involved in implementing case study policies for web browser hardening.

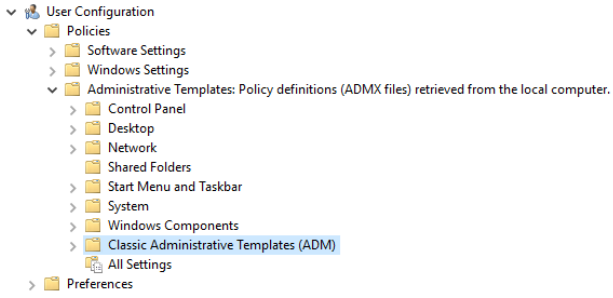

Figure 8. The `User Configuration` group-policy tree.

To make it easier for readers to use this article as a tutorial, we will go through all the steps for remotely hardening each web browser in a separate subsection.

A. Hardening Steps for Internet Explorer

1) Configuration Template Location for Internet Explorer:

Within the group management tree structure, the default location of web browser's policy settings, for domain users, are as follows:

1. In the `Group Policy Management` dialog box, identify the target GPO list. Once found, right-click on it. In the resultant pop-up menu, click `Edit...`.
2. In the `Group Policy Management Edit` dialog, locate the `User Configuration` tree (Figure 9), which can be found in the left pane of the interface. Once located, expand the

Policies tree, located under User Configuration.

3. In the expanded Policies sub-tree, a folder named Administrative Templates should be visible; expand it. In the resulting tree view, a sub-tree with the name of Windows Components should be visible; expand it.
4. In the expanded Windows Components sub-tree, a folder named Internet Explorer should be visible; Open it by double-clicking on the folder name.

2) Implementing the Hardening Policy for Internet Explorer:

1. Navigate to the location of Internet Explorer policies (Sub-Section 5-A1). In the Internet Explorer folder, search for a folder named Internet Control Panel. Once found, double-click on it. Search for a folder Security Page.
2. In the Security Page folder, search for a policy setting Site to Zone Assignment List. Once found, double-click on it. In the resultant box, select the radio button Enabled. Doing so will activate the Show...-labeled button under Options pane, beside the string Enter the zone assignments here. Click on the Show... button.
3. A Show Contents box will pop-up. In this box, under the Value name column, type-in the desired web domain, in this case: www.uidaho.edu and under the Value column, type-in the number 1. Then click OK. This assigns the web domain www.uidaho.edu and all its' sub-domains to the Intranet zone of Internet Explorer.
4. Back in Security Page folder, search for a policy setting Internet Zone Template. Once found, double-click on it. In the resultant box, select the radio button Enabled. Doing so will activate a dropdown menu under Options pane, beside the string Internet. Select the option High from the dropdown menu. Then click OK.
5. Back in Security Page folder, search for a folder Internet Zone. Once found, double-click on it. Inside Internet Zone folder, click on Setting column in the top of middle pane. Doing so will categorize all policy settings into ascending order. Set all policy settings starting with the characters A, D, and J, to the value Disable. Disabling these policy settings will disable most functionality of untrusted websites. Only text would be visible, thereby neutralizing any harmful script- or image-based attacks.

B. Hardening Steps for Microsoft Edge

1) Importing the Configuration Template for Microsoft Edge:

1. Download the administrative template installer for Microsoft Edge [13]. Install the template package to a location of preference.
2. By default, the package is installed at C:\Program Files\Microsoft Group Policy\Windows 10 April 2018 Update (1803)\ PolicyDefinitions, for a 64-bit OS (default path verified as of 05/20/2018). Let us call this path as *Default Install* pathname.
3. Navigate to the *Default Install* pathname's Policy Definitions folder and within this folder, identify the file with name MicrosoftEdge.admx. Once found, copy the file to C:\Windows\ PolicyDefinitions.
4. Navigate back to the Default Install pathname's Policy Definitions folder and within this folder, open the folder matching the OS's locale. In our case, the folder name will be en-US. Within the en-US folder, identify the file with name MicrosoftEdge.adml. Once found, copy the file to C:\Windows\PolicyDefinitions\en-US, or as corresponds to the OS locale.
5. In the Group Policy Management dialog box, identify the target GPO list. Once found, right-click on it. In the resultant pop-up menu, click on Edit... option.
6. In the subsequent Group Policy Management Editor interface, locate the Computer Configuration tree, which can be found in the left pane of the interface. Once located, expand the Policies tree, located under Computer Configuration.
7. In the subsequent expanded tree view, a tree with name of Administrative Templates should be visible. Expand it. In the subsequent expanded tree view, another tree with the name of Windows Components should be visible. Expand it. In the subsequent expanded tree view, a folder with the name of Microsoft Edge should be visible. Open it (by double-clicking on the folder name).

8. If the left side of bottommost bar of the Group Policy Management Editor shows 39 settings (as of 05/20/2018), and the policy settings are visible in the central pane of Group Policy Management Editor interface, then the import process can be considered as successful.

2) Configuration Template Location for Microsoft Edge:

The steps 1-3 are same as stated for Internet Explorer in (5-A1).

4. In the subsequent expanded tree view, a folder with the name of Microsoft Edge should be visible. Open it by double-clicking on the folder name.

3) Implementing the Hardening Policy for Microsoft Edge:

1. Navigate to the default location of Microsoft Edge policies (Sub-Section 5-B2). In the Microsoft Edge folder, search for a policy setting named Configure cookies. Once found, double-click on it.
2. In the resultant box, select the radio button Enabled. Doing so will activate the Configure Cookies labeled drop-down menu in the Options pane. In the drop-down menu, select the option Block only 3rd-party cookies. Afterwards, click on OK button.
3. Due to the nascent status of Microsoft Edge's group policy support, there are no other policies which can contribute towards disabling vulnerable content on a specific web domain, while allowing the content on trusted web domains. However, the following are some policies which might harden Microsoft Edge against web-based attacks at the expense of losing some functionality; irrespective of web domains.
4. In the Microsoft Edge folder, search for a policy setting named Allow Adobe Flash. Once found, double-click on it. In the resultant box, select the radio button Disabled. Afterwards, click on OK button.
5. Back in the Microsoft Edge folder, repeat the same process of step 4 policy settings named Allow search engine customization, Allow web content on New Tab page, Allow Developer Tools, Allow Extensions, Configure additional search engines, and Configure Start pages.

All the modified configuration-policy settings will be in effect upon restart or re-log of domain user accounts. Alternatively, system administrators can force group-policy update for all entities of an OU; by right-clicking on the OU in Group Policy Management dialog box and in the resultant pop-up menu, select the option Group Policy Update.... Provide confirmation in the resultant Force Group Policy update box, by clicking on the Yes button.

C. Hardening Steps for Google Chrome

1) Remote Installation of Google Chrome:

The following steps specify how to remotely install Google Chrome in Windows clients.

1. Download the Google Chrome enterprise installer, depending on the client target OS and architecture from [14].
2. Place the installer in a network-accessible folder that can be read by all target clients. Ideally, such folder must be read-only under the role used by the installer.
3. In the Group Policy Management interface box, identify the target GPO list. Once found, right-click on it. In the resultant pop-up menu, click on the Edit... option.
4. In the subsequent Group Policy Management Edit interface, locate the Computer Configuration tree, which can be found in the left pane of the interface. Once located, expand the Policies tree, located under Computer Configuration.
5. In the subsequent expanded tree view, a folder with name of Software Settings should be visible. Double-click on the folder. Subsequently, a folder named Software installation should be visible. Right-click on it.
6. In the resultant pop-up menu, hover mouse pointer on the New selection. A sub-menu item named Package... should pop up. Click on it. A directory browser will pop-up. Navigate/browse to the location of the Google Chrome installer. Select the installer and click on the Open button.
7. Subsequently, the Deploy Software dialogue box will pop-up. The default selection in this box should be Assigned radio button. If not, change it to Assigned and click on OK.
8. All systems connected to the domain will run the installer upon restart or log-on of domain users. Alternatively, a system administrator can also right-click on a target OU and in the resultant pop-up menu, select Group Policy

`Update...` option to execute the installation right away.

2) Importing the Configuration Template for Google Chrome:

1. Download the Administrative Templates (ADM) package for Google Chrome [15]. Extract the template packages to a location of preference.
2. In the `Group Policy Management` interface box, identify the target GPO list. Once found, right-click on it. In the resultant pop-up menu, click on `Edit...` option.
3. In the subsequent `Group Policy Management Edit` interface, locate the `Computer Configuration` tree, which can be found in the left pane of the interface. Once located, expand the `Policies` tree, located under `Computer Configuration`.
4. In the subsequent expanded tree view, a folder with name of `Administrative Templates` should be visible. Right-click on the folder. In the resultant pop-up menu, click on `Add/Remove Templates...` option.
5. In the subsequent `Add/Remove Templates` pop-up, click on `Add...` button. A directory browser will popup consequently. Navigate/browse to the location of extraction for Google Chrome administrative templates package. Within the extracted package folder, navigate to the preferred locale folder, example: `en-US`.
6. A file named `chrome.adm` (filename verified as of 05/20/2018) will be found within any locale folder. Select the file and click on `Open` button, which can be found at the bottom of navigation window.
7. Back in `Add/Remove Templates` pop-up, click on `Close` button. As a result, a new folder with the name `Classic Administrative Templates (ADM)` will be added, within `Administrative Templates` folder. Double-click on this newly added folder.
8. Double-click on the resultant `Google` folder and the two folders containing configuration policies for Google Chrome web browser can be found in this location. One folder's policies can be overridden by standard users and the other folder's cannot be overridden by standard users.

3) Configuration Template Location for Google Chrome:

The steps 1-2 are same as stated for Internet Explorer in (5-A1).

3. In the subsequent expanded tree view, a tree with name of `Administrative Templates` should be visible. Expand it. In the subsequent expanded tree view, another tree with the name of `Classic Administrative Templates (ADM)` should be visible. Expand it.
4. In the subsequent tree view, a folder named `Google` should be visible; Expand it. Resultantly, there should be two folders visible, with the names `Google Chrome` and `Google Chrome - Default Settings (users can override)`.
5. For this tutorial, we will be focusing on `Google Chrome` folder settings, so that users

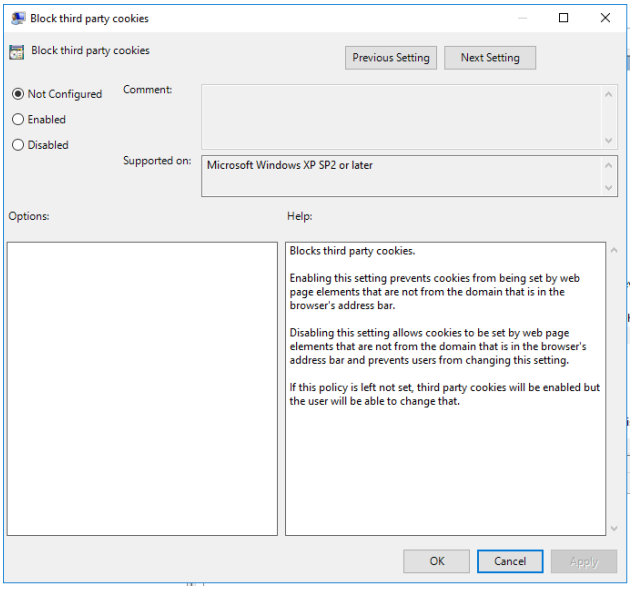

Figure 10.The dialog window for `Block third party cookies` configuration for Google Chrome.

cannot override organizational configurations. As such, open the `Google Chrome` folder (by double-clicking on the folder name).

4) Implementing the Hardening Policy for Google Chrome:

1. Navigate to the default location of Google Chrome policies (Sub-Section 5-C3). In `Google Chrome` folder, search for a folder named `Content Settings`. Double-click on it. Now, search for a policy setting named `Allow JavaScript on these sites`. Once found, double-click on it.
2. In the resultant box, select the radio button `Enabled`. Doing so will activate the `Show...`-labeled button under `Options` pane, beside the string `Allow JavaScript on these sites`. Click on the `Show...` button.
3. A `Show Contents` box will pop-up. In this box, under the `Value` column, type the domain name pattern `[*.]example.org` to white-list JavaScript functionality for all subdomains of *example.org/*. Then click `OK`.
4. Back in `Content Settings` folder, search for a policy setting named `Block cookies on these sites`. Once found, double-click on it. In the resultant box, select the radio button `Enabled`. Doing so will activate the `Show...`-labeled button under `Options` pane, beside the string `Block cookies on these sites`. Click on the `Show...` button. A `Show Contents` box will pop-up. In this box, under the `Value` column, type the organizational web domain name pattern, for example - `[*.]uidaho.edu`.
5. Repeat the above step for all policy settings starting with character `B` (as of 05/20/2018) in the `Content` folder. Since the policy being implemented forbids Google Chrome from connecting to any trusted web-domain, disabling these policy settings will disable most functionality of trusted web domain, thereby not allowing users to go against the organizational policy.
6. Back in `Google Chrome` folder, search for a policy setting named `Block third party cookies`. Once found, double-click on it. In the resultant box, as seen in Figure 10, select the radio button `Enabled`. Then click `OK`. This policy setting will disable third party cookies across all we domains.

6. Practical and Theoretical Implications

The contribution presented in this article leads to the following theoretical and practical implications.

A. Theoretical Implications

The perimeter-based security or walled-castle approach, which has been applied towards protecting the enterprise, has been successful at preventing most cyber-attacks that are launched from the outside. Today, from an attacker's point-of-view, the path of least resistance is tricking users to launch an attack from within the security perimeter. The walled-castle approach to cyber-defense is necessary, but no longer sufficient in securing an organization's data and infrastructure. Time has come to secure the enterprise at the role-, user-, device-, and application-levels. This can be accomplished through high-granularity and tailored security configurations. Today, the weakest and highest targeted applications are email and internet browsers. For this reason, we have prioritized developing techniques and tools for securing the client browser ecosystem.

We argue that this implication also applies to all applications being utilized in an enterprise, not just web browsers. This realization helped shape our understanding of this project's current and future practical implications.

B. Practical Implications

Currently, high-granularity and tailored security configurations may be performed manually across some enterprise applications. A detailed process for the manual configuration approach needed for hardening web browsers in the enterprise is the main contribution of this article. However, the efficiency, sustainability, scalability, and accuracy of such hardening processes is being heavily compromised by the lack of automated and policy-based design and deployment technologies and tools.

To tackle this challenge, we are currently working on developing a tool-set to help system administrators design, and automatically deploy, secure application configuration policies at an enterprise scale. This tool-set is called HiFiPol:Browser and is policy-based and independent of the platform, application, device, user, and role aspects of an organization. Additional information on the HiFiPol:Browser project may be found in Section 7 of this article.

7. Related Contributions

In previous publications [24], [25], [26], we describe a prototype system, HiFiPol:Browser, for policy-based and enterprise-scale system hardening. With HiFiPol:Browser, system administrators can use a high-level, English-like specification language to represent their organizational infrastructure and security policies.

This specification language is called HERMES and it is independent of platforms, applications, users, roles, and devices being configured. HiFiPol:Browser uses HERMES policy specifications to remotely configure client-side applications as per the specifications provided in the policies..

HiFiPol:Browser is in an early prototype stage and we expect that it will be several years before the system is ready for commercial enterprise deployment. Hence, we decided to put forth the contribution in this article hoping that, with the current tools, organizations can still deploy, today, a hardened browser infrastructure, even if at a higher configuration and maintenance cost than when compared with automated policy-level tools, which we are currently working on designing and developing.

8. Related Remote Configuration Resources

There are several online articles, which detail how to setup and use ADDS. We have written a step-by-step tutorial about the setup process of an ADDS domain and assigning a controller to the domain [12]. Microsoft has published multiple articles which guide deployment, operations, and troubleshooting of ADDS infrastructure [16], [17]. There are also several books and online articles which explain the process of managing an organization's systems using Group Policies. Jeremy Moskowitz has authored a book, which explains in detail, all processes involved in using group policy tools including administrative templates [18]. Microsoft has also published a beginner's overview tutorial for `Group Policy Management tool` [19].

There are also many articles describing how to change a given configuration option for a given web-browser, using group policies in an ADDS infrastructure. Google has published multiple and detailed guides, which describe about: a) ADDS-based enterprise-level remote deployment of Google Chrome [20], and b) setting Google Chrome group policies for devices, and users [21]. Similarly, Microsoft has also published brief group policy setting guides for both Internet Explorer [22] and Microsoft Edge [23].

However, we did not find a standalone step-by-step tutorial, that would guide a system administrator through the process of instantiating a high-level security policy that spans multiple applications, into the corresponding security configurations. Currently to do this, a system administrator must: a) read all the existing tutorials, b) assemble all the pieces of knowledge together, and c) fill-in the gaps between knowledge pieces by conducting research. In this article, we present a standalone step-by-step tutorial that details the process of instantiating a multi-application high-level security policy into corresponding security configurations. We do this with the use of a case study which implements least privilege web browsing policy.

9. Conclusion

In this tutorial article, we described, step-by-step, how to use Microsoft's Active Directory Domain Services and Group Policy to remotely configure Internet Explorer, Microsoft Edge, and Google Chrome to implement a specific Least Privilege security policy. In such policy, a hardened browser (Google Chrome) is dedicated to browsing the Web at-large, and a second browser (Internet Explorer or Microsoft Edge) is restricted to accessing Intranet sites. ADDS is the State-of-the-Practice and the most widely-used system in the enterprise for configuring and managing Windows-based clients. We hope that, by following and adapting this tutorial, organizations begin to configure their client browsing infrastructure with hardened configurations that implement a least privilege policy. We believe that hardening browsers, in addition to the network and server sides, will help prevent and mitigate the current prevalence of browser-based cyber-attacks.

10. Future Work

The research problem of automatic and remote deployment of configurations through high-level security policy involves the following steps: 1) generalizing the enforcement of security policies, 2) automating the process of enforcing and compliance check, 3) verifying the enforcement/compliance of security policies, and 4) evaluating the effectiveness of this method via experimental studies. To be specific, the research problem requires fulfilling the following contributions:

1. Present the complexity of high-level policy enforcement in web browsers, by explaining the inefficiency of current tools in enforcing high-level policies. Demonstrate in a step-by-step manner on how to enforce a high-level policy in web browsers using current tools, instead of waiting for the development of a new tool. (This is the contribution that we are reporting about in the submitted manuscript).
2. Architecture design of a tool for policy enforcement; (We designed a tool called HiFiPol:Browser and reported about it in a conference publication [25]).
3. High level policy language for configuration specification; (We developed a language called HERMES and reported about it in a conference publication [26]).
4. Design of organizational case studies;
5. Design and develop policy conflict detection algorithms;
6. Design and develop a platform for semi-automatic or automatic policy instantiation, which transforms the high-level policy into low level configurations;

7. Design and develop a configuration repository, which stores configuration files to be implemented;
8. Evaluate various deployment platforms and select one which is best suited/applicable for deploying web browser configuration files;
9. Evaluate the effectiveness and/or usefulness of this method via experimental studies using volunteer user groups.

LIST OF ABBREVIATIONS

AD - Active Directory;
ADM - Administrative Templates;
ADDS - Active Directory Domain Services;
ADML - Administrative Template, localized configuration description file;
ADMX - Administrative Template, configuration file;
GPO(s) - Group Policy Object(s);
GC - Google Chrome;
HERMES - High-level, Easy-to-use, and Re-configurable Machine Environment Specification language;
HiFiPol - High Fidelity and Policy oriented Tool-set;
IE - Internet Explorer;
ME - Microsoft Edge;
OU(s) - Organizational Unit(s);
VM(s) - Virtual Machine(s);
URL - Uniform Resource Locator.

ACHKNOWLEDGMENTS

We would like to thank the U.S. National Science Foundation (NSF) and the State of Idaho, for funding this research work. NSF under CyberCorps awards 1027409 and 1565572. Idaho under IGEM Cybersecurity grants of 2012 and 2016. The funding bodies were not involved in any phase of this research (study, design, experiments, and writing), other than providing the funding. The opinions expressed in this paper are not necessarily those of the U.S. Government or the State of Idaho. The U.S. Government and the State of Idaho reserve the right to copy and distribute this article as needed for Government purposes.

We would like to thank all the personnel at the University of Idaho's Information Technology Services, College of Engineering, Computer Science Department, and Center for Secure and Dependable Systems for maintaining our research and operational infrastructure and operations. We would also like to thank Mr. Victor House for contributing to our discussions about information infrastructure security. We would also like to thank the journal editors and manuscript reviewers for their time evaluating this article and their constructive reviews.

REFERENCES

[1] Wikipedia, "Turing completeness," Online, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Turing completeness

[2] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," Proceedings of the IEEE, vol. 63, no. 9, pp. 1278– 1308, 1975.

[3] Verizon, Inc., "2016 data breach investigations report," Online, pp. 22–23, April 2016. [Online]. Available: www.verizonenterprise.com/ resources/reports/rp DBIR 2016 Report en xg.pdf

[4] Verizon Inc., "2017 data breach investigations report," Online, June 2017. [Online]. Available: www.verizonenterprise.com/resources/reports/rp DBIR 2017 Report en xg.pdf

[5] R. Joyce, "Disrupting nation state attacks," in USENIX Enigma Conference 2016. [Online]. Available: https://www.youtube.com/watch? v=bDJb8WOJYdA

[6] A. A. Jillepalli, D. Conte de Leon, S. Steiner, F. T. Sheldon, and M. A. Haney, "Hardening the client-side: A guide to enterprise-level hardening of web browsers," in Proc. 2017 of 15th IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC2017), November 2017.

[7] Microsoft TechNet, "Active directory domain services overview," Online, April 2007. [Online]. Available: https://technet.microsoft.com/ en-us/library/9a5cba91-7153-4265-adda-c70df2321982

[8] Microsoft Corp., "Group policy management console," Online, 2017. [Online]. Available: https://technet.microsoft.com/en-us/library/cc753298

[9] J. Herman, "Managing group policy ADMX files step-by-step guide," Online, Microsoft, Technet, 2007. [Online]. Available: https: //msdn.microsoft.com/en-us/library/bb530196.aspx

[10] Microsoft TechNet, "Active directory and active directory domain services port requirements,". [Online]. Available: https: //technet.microsoft.com/en-us/library/dd772723

[11] Killerxcana and Waslow, "Gpo for firefox add-on,". [Online]. Available: https://addons.mozilla.org/en-US/firefox/addon/gpo-for-firefox/

[12] A. Jillepalli and N. Nuthalapati, "Domain controlling: Group policy with active directory," https://goo.gl/kEj31U, May 2016.

[13] Microsoft Corp., "Microsoft edge administrative templates,". April 2018. [Online]. Available: https://www.microsoft.com/en-us/ download/confirmation.aspx?id=56880

[14] Google Inc., "Google chrome enterprise installers," Online, 2018. [Online]. Available: https://enterprise.google.com/chrome/chrome-browser/

[15] Google Inc., "Google chrome administrative templates," online, February 2018. [Online]. Available: https://dl.google.com/dl/edgedl/ chrome/policy/policy templates.zip

[16] Microsoft TechNet, "Active directory domain services for windows server," 2009. [Online]. Available: https://technet.microsoft.com/en-us/library/dd378891(v=ws. 10).aspx

[17] Microsoft Corp., "Introduction to active directory domain services (ad ds) virtualization (level 100)," 2017. [Online]. Available: https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/introduction-to-active-directory-domain-services-ad-ds-virtualization-level-100

[18] J. Moskowitz, "Group policy: Fundamentals, security, and the managed desktop," 2015. [Online]. Available: http://www.wiley.com/WileyCDA/ WileyTitle/productCd-1119035589.html

[19] Microsoft TechNet, "Group policy for beginners," 2011. [Online]. Available: https: //technet.microsoft.com/en-us/library/hh147307(v=ws.10).aspx

[20] Google Inc., "Chrome deployment guide," 2018. [Online]. Available: https://goo.gl/CNzDA5

[21] Google Inc., "Configure policies and settings," 2018. [Online]. Available: https:// support.google.com/chrome/a/topic/4386995?hl=en&ref topic=4386754

[22] Microsoft Corp., "Administrative templates and internet explorer 11," 2018. [Online]. Available: https://docs.microsoft.com/en-us/internet-explorer/ ie11-deploy-guide/administrative-templates-and-ie11

[23] Microsoft Corp., "Group policy and mobile device management (mdm) settings for microsoft edge," 2018. [Online]. Available: https://docs.microsoft.com/ en-us/microsoft-edge/deploy/available-policies

[24] D. Conte de Leon, V. A. Bhandari, A. A. Jillepalli, and F. T. Sheldon, "Using a knowledge-based security orchestration tool to reduce the risk of browser compromise," in Proc. 2016 IEEE 07th Symposium Series On Computational Intelligence (SSCI-2016), December 2016.

[25] A. A. Jillepalli and D. Conte de Leon, "An architecture for a policyoriented web browser management system: HiFiPol: Browser," in Proc. 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC-2016), June 2016.

[26] A. A. Jillepalli, D. Conte de Leon, S. Steiner, and F. T. Sheldon, "Hermes: A high-level policy language for high-granularity enterprisewide secure browser configuration management," in Proc. 2016 IEEE 07th Symposium Series On Computational Intelligence (SSCI-2016), December 2016.

Ananth A. Jillepalli. Ananth is a PhD Candidate at the University of Idaho. He received his Master of Science in Computer Science (Cybersecurity) degree from University of Idaho in 2017. His current research interests include design and analysis of risk assessment methodologies and relevant applications.

Daniel Conte de Leon. Dr. Conte de Leon is a cybersecurity researcher and educator at the University of Idaho. He received his PhD degree in Computer Science in 2006 from the University of Idaho. Dr. Conte de Leon performs research on the development of processes, methods, and tools for the design, development, configuration, and maintenance of safe and secure systems.

Frederick T. Sheldon. Prof. Sheldon has 35+ years in the fields of software engineering and computer science engaged as an engineer, principal investigator, research scientist, business developer and academic administrator. He's held faculty appointments at U. Colorado CS, Wash. State U., U. Memphis, CS Chair at U. Idaho (current). He's published 150+ articles, 12 editorships, 4 US Patents and chaired and participated in National R&D venues including invited speaker, panelist and moderator.

Michael A. Haney. Dr. Haney is an assistant professor of Computer Science for the University of Idaho and a cybersecurity researcher for the Idaho National Laboratory. Currently, his research interests are in data visualization, specifically visualizing network and system log data to improve intrusion detection and response for large-scale networks. He studies cyber-security issues of energy assurance supporting a more resilient "smart" infrastructure.