Welcome to hardening the Mac OS AND…The Joy of Telling Users No.

Depending on your user base this is either, joy, or 'airquote- joy'.

The more I've gotten involved in the mac admin community, the more I realized that my work environment doesn't match up to a lot of others because of one core component, security hardening. Yet, as Macs become more popular and expand into more places, this is becoming an increasingly important topic.

The question we all ask of course, is why?

The more secure your environment, the stricter you security standards have to be.
- Doctors offices,
- financial services,
- government offices,
- access to a lot of private and often confidential information.

And not only is it important to us personally to protect this data, its the law. We have HIPAA, the Bank Secrecy Act, and PCI Compliance, all requiring us to secure our computers.

HIPAA: federal Health Insurance Portability and Accountability Act of 1996
PCI: The Payment Card Industry Data Security Standard (PCI DSS) is a set of requirements designed to ensure that ALL companies that process, store or transmit credit card information maintain a secure environment.

WHY?

You only have to look towards Target to see that there is also the very real threat of class action lawsuits if/when your security is breached.

As our friend The Hulk says: Security starts from the inside. Start strong. Or HULK SMASH.

PART I
THE SYSTEM

I’ve separated this approach in to 2 sections, the System and the User.

We'll start with the System (because the system can't argue with you).

Creating your standards.

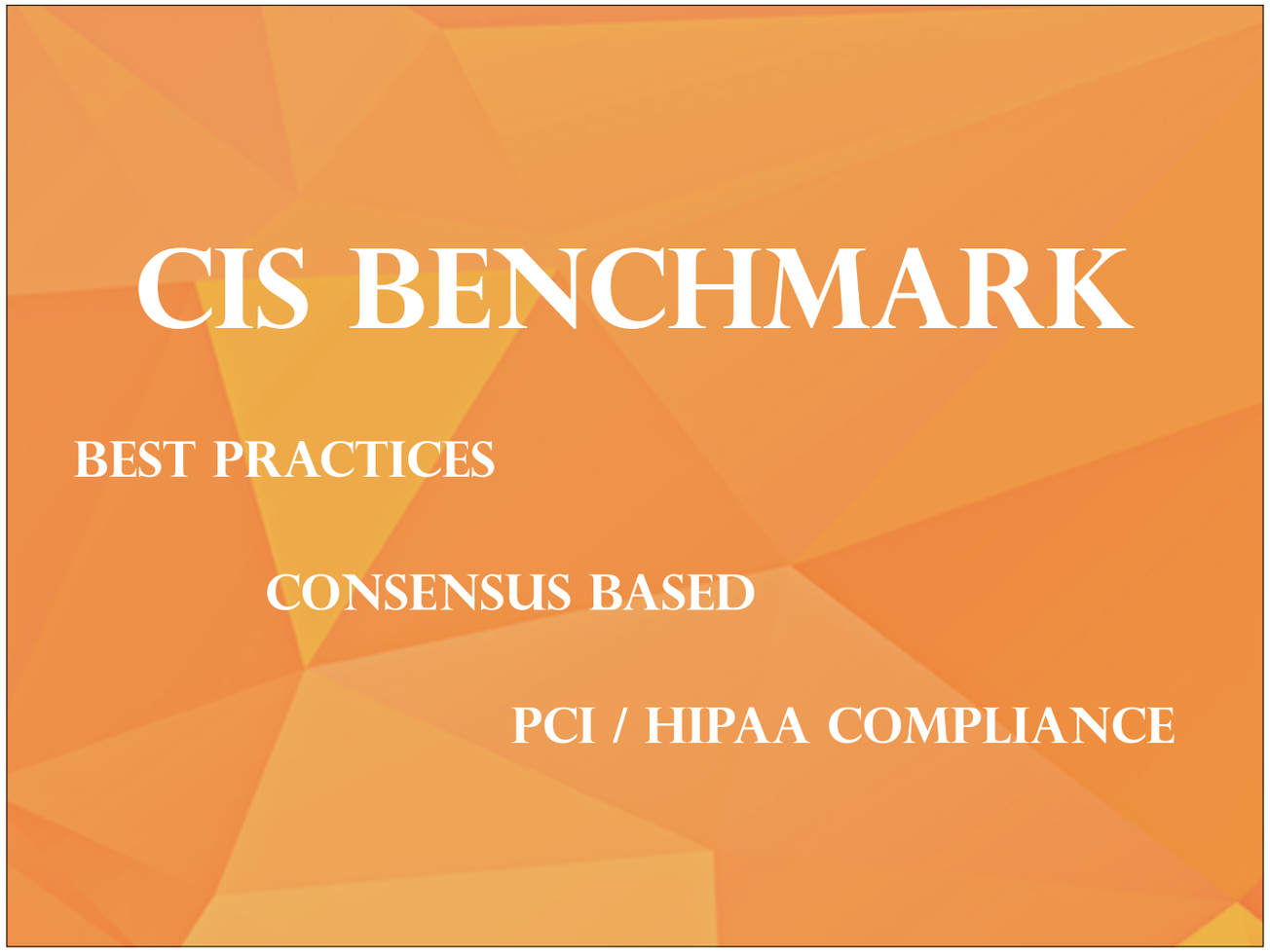
This CANNOT be a one person job.
For many reasons, the first of all being that rarely is it the place of the mac admin to set security rules.

- Expect your other IT departments to look for you for the ‘Mac Equivalent’ of things they are more familiar with on a different OS.
- Expect to be responsible for implementation.

But, deciding on what standards are appropriate for your environment is something that should be discussed on all levels. Its a decision that ultimately should be made by your security departments.

Steps I took in researching security measures for my own environment

- paired with a security specialist and we worked though it together
- creating a set of standards that were then presented to our superiors



To give myself somewhere to start, a basic framework, I like to use the CIS Benchmark documents.

CIS is the Center for Internet Security;

- The Benchmark is a comprehensive document they've put together with recommendations on security items for the operating system.
- it is a collaborative effort of subject matter experts who volunteer their knowledge.
- They have a great reputation and are recommended as industry standard for PCI, HIPAA and other security standards.

2.4 Sharing

This section contains recommendations related to the configurable items under the *Sharing* panel.

2.4.1 Disable Remote Apple Events (Scored)

Profile Applicability:

- Level 1

Description:

Apple Events is a technology that allows one program to communicate with other programs. Remote Apple Events allows a program on one computer to communicate with a program on a different computer.

Rationale:

Disabling Remote Apple Events mitigates the risk of an unauthorized program gaining access to the system.

Audit:

Perform the following to ensure the system is configured as prescribed:

1. Run the following command in Terminal:

```
sudo systemsetup -getremoteappleevents
```

2. Verify the value returned is `Remote Apple Events: Off`

Remediation:

Perform the following to implement the prescribed state:

1. Run the following command in Terminal:

```
sudo systemsetup -setremoteappleevents off
```

Impact:

With remote Apple events turned on, an AppleScript program running on another Mac can interact with the local computer.

This is a sample page of the Benchmark document. It lays out a description, a rationale, an audit method and a remediation action for each recommendation.

It also breaks the recommendation into two profile levels:

Level 1

Items in this profile are intended to:

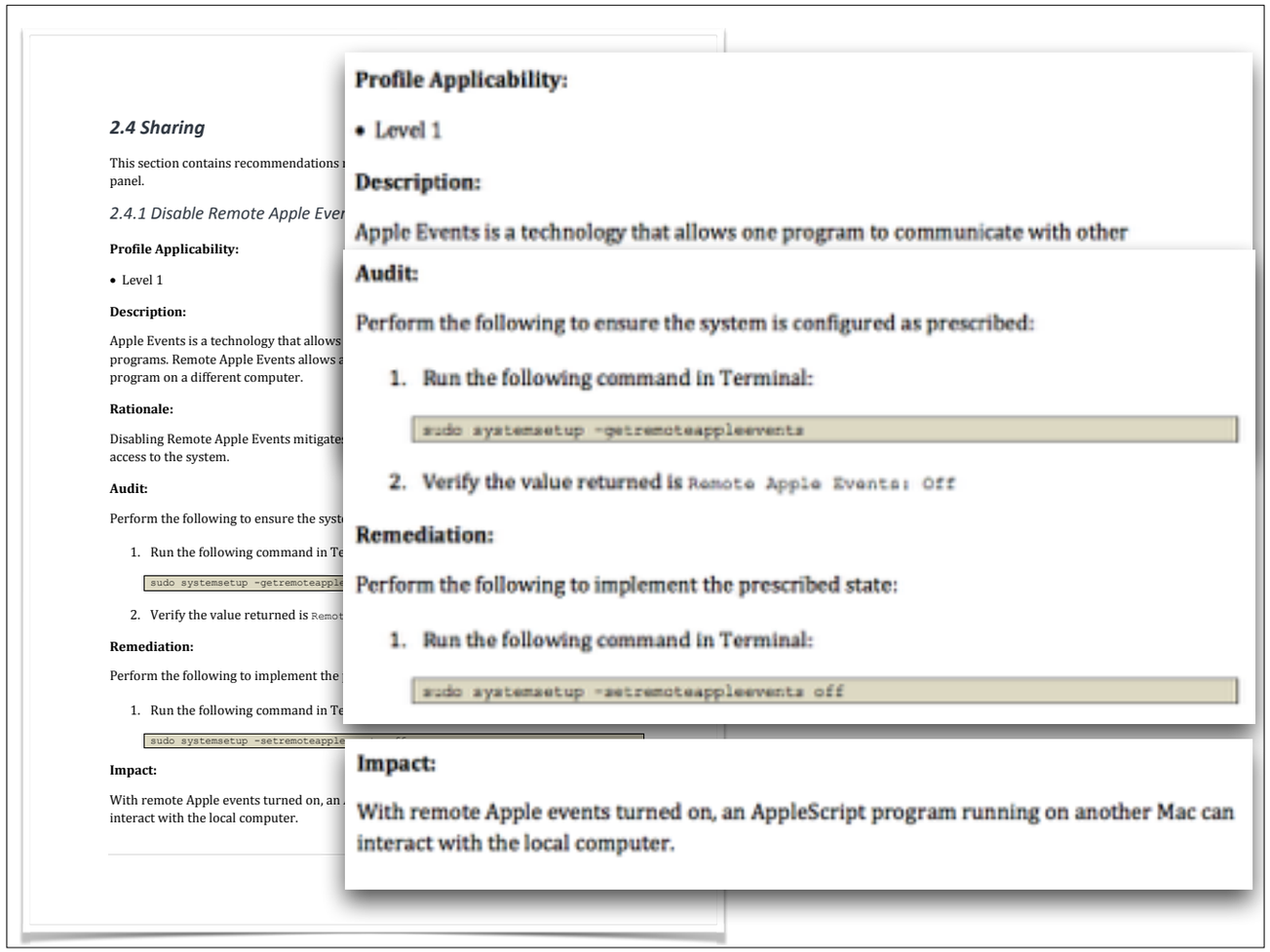
- be practical and prudent;
- provide a clear security benefit; and
- not inhibit the utility of the technology beyond acceptable means.

Level 2

This profile extends the "Level 1" profile.

Items in this profile

- are intended for environments or use cases where security is paramount
- acts as defense in depth measure
- and/or may negatively inhibit the utility or performance of the technology.



2.4 Sharing

This section contains recommendations r
panel.

2.4.1 Disable Remote Apple Ever

Profile Applicability:

- Level 1

Description:

Apple Events is a technology that allows
programs. Remote Apple Events allows a
program on a different computer.

Rationale:

Disabling Remote Apple Events mitigates
access to the system.

Audit:

Perform the following to ensure the syst

1. Run the following command in Te

```
sudo systemsetup -getremoteapple
```

2. Verify the value returned is `Remot`

Remediation:

Perform the following to implement the

1. Run the following command in Te

```
sudo systemsetup -setremoteapple
```

Impact:

With remote Apple events turned on, an
interact with the local computer.

Profile Applicability:

- Level 1

Description:

Apple Events is a technology that allows one program to communicate with other

Audit:

Perform the following to ensure the system is configured as prescribed:

1. Run the following command in Terminal:

```
sudo systemsetup -getremoteappleevents
```

2. Verify the value returned is `Remote Apple Events: Off`

Remediation:

Perform the following to implement the prescribed state:

1. Run the following command in Terminal:

```
sudo systemsetup -setremoteappleevents off
```

Impact:

With remote Apple events turned on, an AppleScript program running on another Mac can interact with the local computer.

For example, you can see here the recommendation to Disable Remote apple events. It is a level 1 Item.

- It explains what 'Remote Apple Events' are:
  (Apple Events is a technology that allows one program to communicate with other programs. Remote Apple Events allows a program on one computer to communicate with a program on a different computer.)

- Next it tells us why we should think about disabling them,
  (Disabling Remote Apple Events mitigates the risk of an unauthorized program gaining access to the system.)

- Third, you are given directions for checking to see if it is implemented on your system,
  here it is a terminal command, sudo system setup -getremote appleevents

- And then you are shown how to enable the setting, should you decide it is applicable.
  here it is the same terminal command, adding the OFF flag to the value you read in the audit command.

- Finally, the document explains what impact this change will have on your system. And what kind of security exposure may exist.
  With remote Apple events turned on, an AppleScript program running on another Mac can interact with the local computer.

CIS Benchmark 10.9

- Configure security auditing flags
- Retain install.log for 365 or more days

- Level 2
  - Remote logging for desktops on trusted networks
- Network Configurations
  - Level 1
    - Show Wi-Fi in menu bar
  - Level 2
    - Disable Bonjour advertising
    - Create Network specific locations
- System access authentication and authorization
  - Level 1
    - Secure home folders (top level)
    - Repair Permissions regularly
    - Check System Wide Applications for appropriate permissions
    - Check system and library folders for world writeable
    - Auto lock keychain after 5 minutes and when sleeping
    - Do not enable root account
    - Require password to wake from sleep or screen saver
    - Disable Auto login
    - Require an administrator password to access system-wide preferences
    - Disable ability to login to another users active and locked session
    - Complex passwords
    - Account Lockout
    - Access warning for login window
    - No password hints
  - Level 2
    - Disable fast user switching
    - Secure individual keychain items
    - Create specialized keychains for different purposes
- User Accounts and Environment
  - Level 1
    - ~~Login window as name and password~~
    - Disable show "password hints"
    - Disable Guest account
    - Disable guest access to shared folders
    - Turn on filename extensions
    - Disable automatic run of safe files in safari
    - User parental controls for systems not centrally managed
- Additional Considerations
  - iCloud Configuration
  - Wireless Adapters
  - iSight Camera
  - Computer Name considerations
  - Software Inventory Considerations
  - Firewall Consideration
  - Automatic Actions for Optical media
  - App Store Automatic Downloads
  - EFI Passwords

would need a location
team security to steer

default

only one - internal

Verify this is the time we want

Research This one!

contradicts filevault encrypt

exceptions for kiosk

blocked on network level

not in use

off per company policy

via company pol

no admin privileges to users

no drives

App store blocked

already in use

external firewalls and network security maintained by other team

AD controlled

AD controlled

As I said, The CIS benchmark is obviously not the end all, be all, of Mac OS X security requirements.

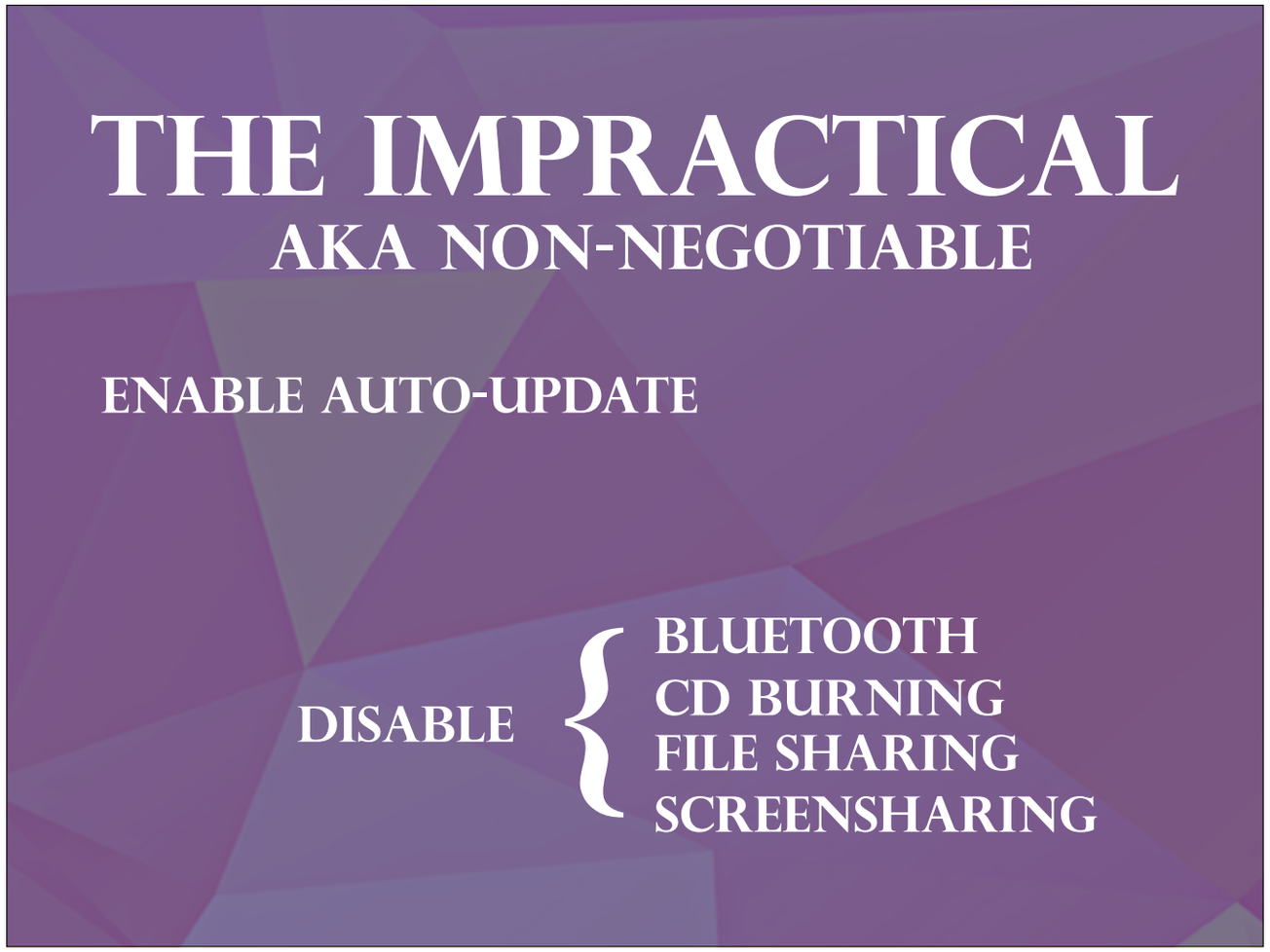
- good jumping off point,
  - especially if you are struggling to figure out where to begin.
- Reference point for documentation and the approvals process.

This document is 99 pages, so I made my self an outline so it was an little easier to work with. Everyone evaluates these types of things differently, this is how i decided to do it.

My outline....
Once I took at look at it, and scribbled all over it, I was able to determine

- what we were already doing,
- what needed to be looked into further
- (and what needed to be ignored).

I chose five categories.
Things that just don’t apply to me,
Things I was going to do anyway,
Things that make security happy,
Things we should talk about
and then Things I felt they left off the list.

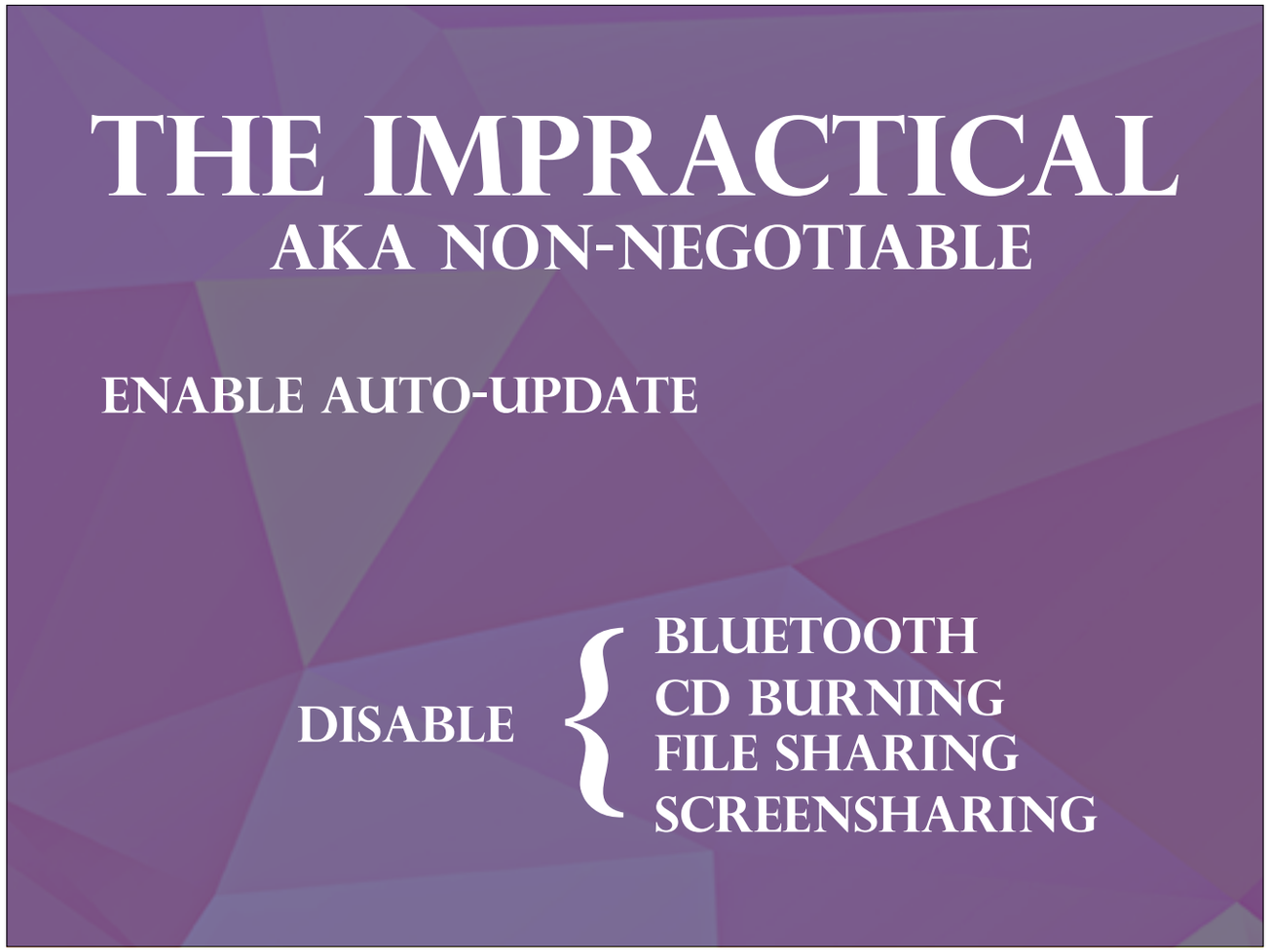


First, the Impractical
These are those items that you know from the beginning simply don't apply to your organization.

environment non-negotiable

You'll hear me say this a lot…these will vary for everyone…because it will. As Mac admins our companies may all have different backgrounds, so we have to make individual decisions on many of these security items.

For example: auto updates.
We all try to find the balance between keeping up with security patches, while throughly testing updates for bugs or conflicts before they are deployed. You may or may not want this activated.
One option is having your own SUS, software update server, along with a solid update plan/schedule. This way you can deploy patches once they've been cleared.



continued from previous….

Disabling things: these might be deal breakers for your team.

The CIS Benchmark recommends disabling
- bluetooth
- CD burning
- file sharing
- screen sharing

However,
I can’t use my awesome bluetooth trackpad if bluetooth is disabled
and a photographer can’t very well provide a disk of photo proofs when she can’t burn a CD. Context **really** matters here.

Once you’ve assessed your list and you go back to security, explain what you’ve decided, and why. If possible, show off the Mitigations you’ve put into place to reduce the risk as much as you can.

The Obvious, the things we were going to do anyway.
our “The accepted standards”

Four sections:

Global System Preferences:
Think FileVault, password requirements, keeping the root account disabled

User Preferences:
Wifi/Bluetooth in the menu bar, password hints, screensaver times
Login Window Behavior:
Guest accounts, auto-login, fast user switching, access warnings (ie this computer belongs to the super-secret government agency and anything you do is monitored, etc.)
Gatekeeper
is application white-listing control that restricts downloaded applications from launching.
Disallowing unsigned software will reduce the risk of unauthorized or malicious applications from running on the system.

The Harmonious, these are things that you may or may not care about, but could ultimately make your life easier to have enabled and can improve your relationship with your security team.

Core dumps
Anyone know what this is? (I didn’t when I first read it)

A core dump occurs when an application encounters a runtime error and the operating system dumps the application's state, including memory contents, to disk.

Rationale:

Since it is possible for a core dump to contain sensitive information, including passwords, it is recommended that core dumps be disabled in high security scenarios.

Update: I learned after the presentation that CoreDumps have been depreciated in Yosemite and Mavericks.

Enabling Secure Keyboard Entry

- minimizes the risk of a key logger from detecting what is entered in Terminal.
- *defaults read command*
- Once enabled, its not likely you’ll even notice it, but again, you’ve provided increased security to the system.

notes continued from previous….

Extend log retention
- allow the user to view the various changes to the system along with the date and time they occurred.
- *modify ttl in etc/asl.conf*
- It gives you the ability to troubleshoot and to AUDIT

Visible filename extensions
- allows the user to identify the file type and the application it is associated with which leads to quick identification of misrepresented malicious files
- *defaults read command*
- Hackers have taken advantage of this setting via drive-by attacks,by auto downloading items from webpages…hoping they will be confused for the items you want, or auto opened by safari

Preventing files from auto-opening in safari,
- you want to make sure that what you are opening is what you really intended on opening.

As our infosec friend Ms. Swift says:

If you get a weird prompt to download something,

Ask yourself, What would Taylor Swift do?

Stop thinking you aren't a target, you are. Start acting like it.

Start Acting like Taylor Swift.

The Negotiable -
These are going to vary from environment to environment.
The important question to remember is, Are you using it?
If so, document your usage.
If not, turn it off.

Example isight and iMessage : enable for (video) chat users only. someone could gain control of the camera

Airdrop & Bonjour: are they really appropriate for an enterprise environment?
You can also customize the settings, on for some, off for others.

App Store Automatically download apps purchased on other Macs
While this feature could be very desirable for personal Macs or a small business setting so that all purchased software through Apple's App Store is provisioned on all OS X Computers, just like iOS. This feature may not be desirable in Corporate environments where the expectations of handling software licenses, tracking software inventory and personal software are different.

The Undocumented

When Yosemite's new features were first announced ,these were some of the things I was immediately concerned about. I knew that security would not want data being shared back and forth from a users personal phone,
which is exactly what continuity was designed to do.

Continuity

- *Continuity refers to the overall thing.*
- OS X Yosemite and iOS 8 enable new features that let your devices work together in even smarter ways.

Handoff

- They can automatically "hand off" what you're doing from one device to another.

Personal hotspot

- Gets you around any network security in place

USB

- file sharing, personal data,

BONUS!

ADMINISTRATOR PRIVILEGES

I debated with myself over which category admin rights belongs in. But its something every admin has to deal with. So I went with a Bonus Item!

The choice to give or deny local admin accounts is a debate I've had with many co-workers and even people in this room. There is rarely a consensus.

Plus, The benchmark doesn't explicitly comment on whether users should have administrator rights or not, but what is does stress is proper account management.

Policies should be in place to ensure that all accounts on the system are the needed accounts, not just the default accounts, and that Administrator functions should be clearly separated from day-to-day functions.
Additionally, user names on accounts with elevated privileges should not be easily guessed by looking at the standard account names.

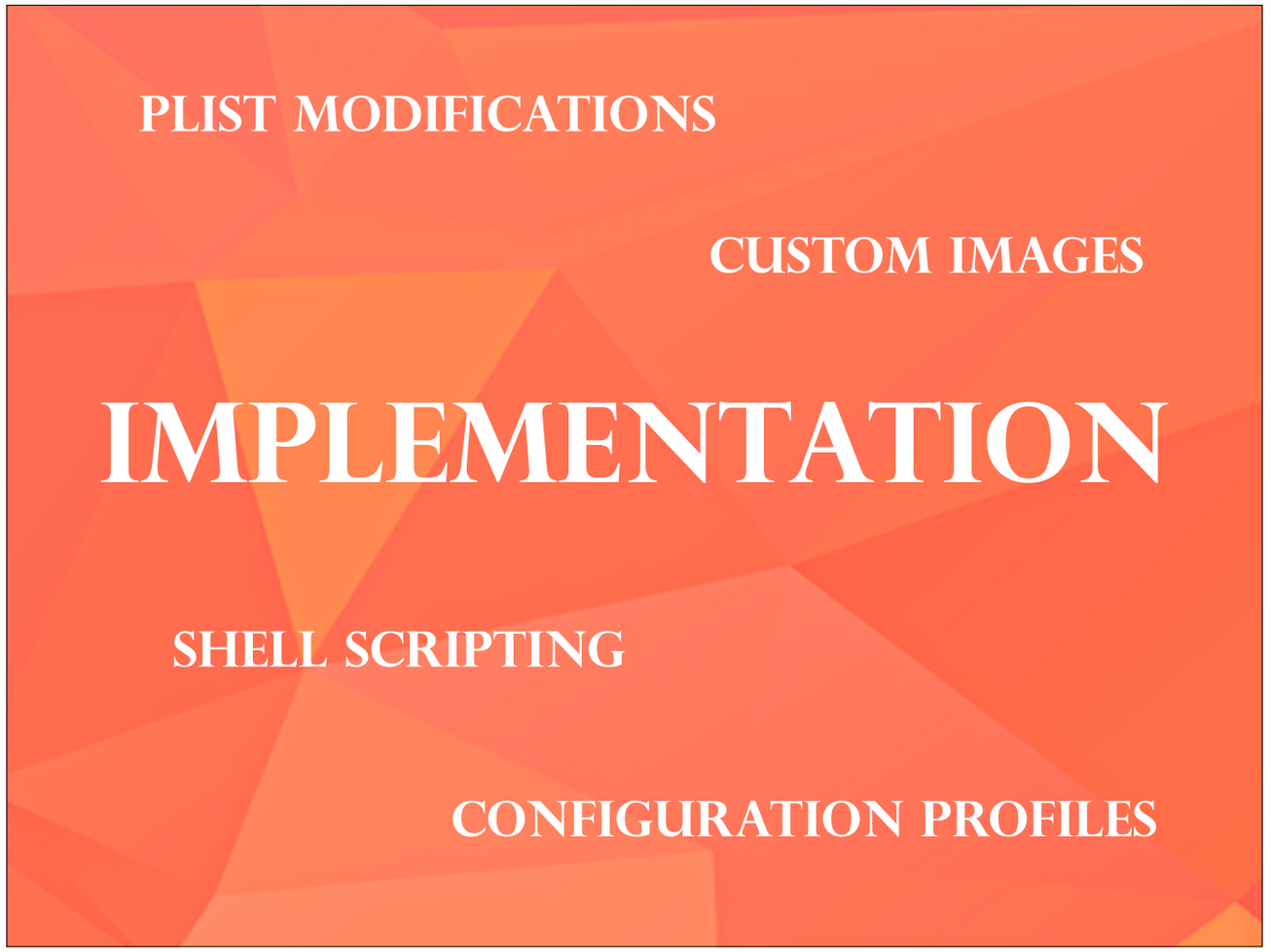
These recommendations are designed to keep your system as secure as possible by providing the least number of ways for someone with malicious intent to access your data.

In 2014, when Rootpipe was reported, it needed admin rights.
Just last month, a flaw was reported that would allow an application access to your keychain and therefore your passwords.
As I'm sure you can guess, It too needed admin rights, to install the initial application from the Mac App store.

And on the other side (windows!) Techworld reported earlier this year that about 97% of vulnerabilities reported for Microsoft systems required Admin rights to exploit.

My point here is this: be careful. with great power comes great responsibility. Think about it…whats the worst that could happen?

(not really adobe's installer, its flashback)



Once I finished working with security on setting the standards, it was left to me to being implementing them.

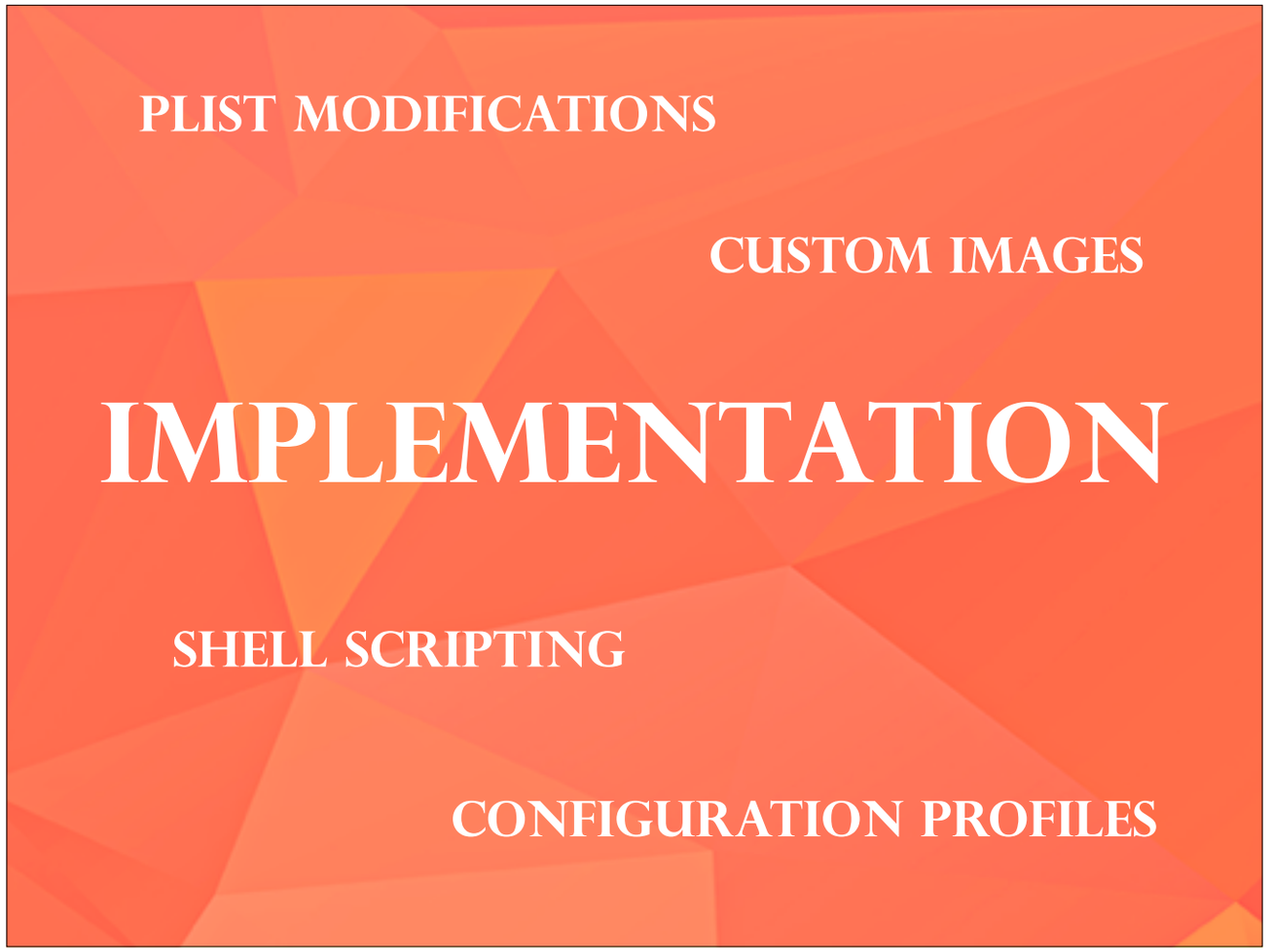
You have choices when it comes to implementing your standards.

plist modifications
- You could make the changes you want to see in your loginwindow.plist
- /Library/Preferences/com.apple.loginwindow.plist
- deploy a copy to all of your machines.

monolithic image for deployment.
- build all of your requirements into the image.

The two we're going to look at a little further are:
shell scripting
- any language you like and your mac can read

Configuration profiles.
- Configuration profiles are XML files that contain preconfigured system settings.
- They are a followup to Apple's managed preferences that can be used with or without MDM (mobile device management).
- And are also in use on the iOS, providing additional commonality for management of both systems.



continued from previous….

I personally use these two because I find them to be more flexible options than plists and monolithic images.

You environment is going to matter just as much for implementation as it did for choosing the standards.

Imagine you walk into an office on your first day and the boss says:
“I need these 50 computers imaged by the end of the week with this big list of security standards. Oh, and we’re sending them halfway across the world without any way to manage them.”
A monolithic image might be just what you need.

“But what if instead he says: those 50 computers are already on the other side of the world and they need to be managed and secured as soon as they touch my network.”
Maybe don’t use a monolithic image in that case.

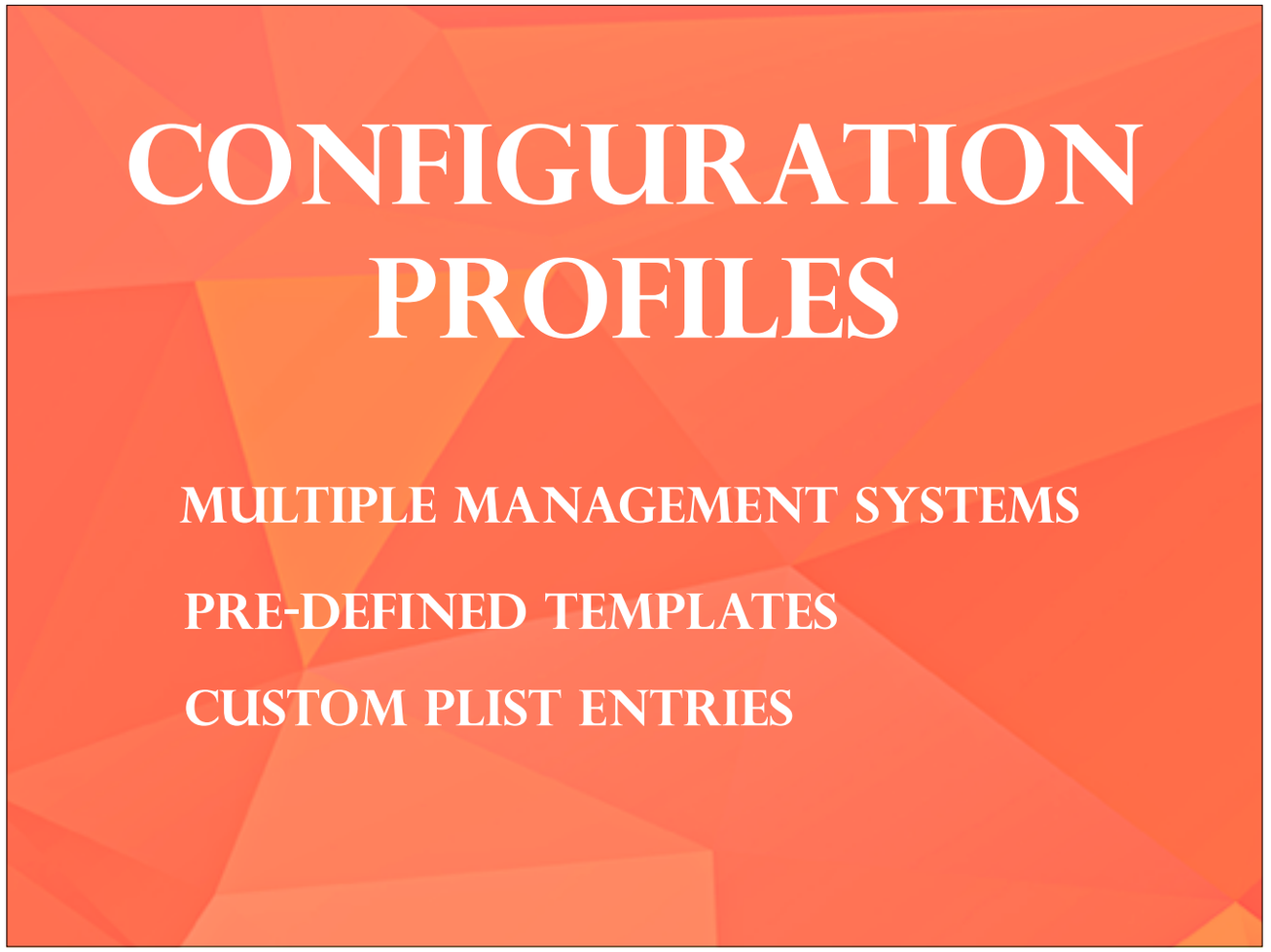
SCRIPTING

```
#!/bin/bash

# Disable iSight

dir=/private/tmp/isight

if [[ ! -e $dir ]]; then

  mkdir $dir

elif [[ ! -d $dir ]]; then

  echo '$dir already exists but is not a directory' 1>&2

fi

mv /System/Library/QuickTime/QuickTimeUSBVDCDigitizer.component/
Contents/MacOS/QuickTimeUSBVDCDigitizer /private/tmp/isight

mv /System/Library/Frameworks/CoreMediaIO.framework/Versions/A/
Resources/VDC.plugin/Contents/MacOS/VDC /private/tmp/isight

mv /Library/CoreMediaIO/Plug-Ins/DAL/AppleCamera.plugin/
Contents/MacOS/AppleCamera /private/tmp/isight
```

Scripting is an option for admins on multiple skill levels, because
a) there is an awesome Mac community online willing to share
and
b) you don't have to be super adept at coding to get something done.

your script can be as simple as one or two lines of commands, as you can see here in this example.
i'm just moving 3 files to a hidden tmp location. Moving these 3 files effectively turns off the iSight Camera, and I've moved them, instead of deleting them so I can put them back later if I need to.

Now, do I write great scripts? No. Could this be written better? Probably. Does this work? Absolutely.



We could easily fill a presentation on configuration profiles, in the interest of time, for a full overview on configuration profiles, I'm going to point you to a few links I've included in the 'resources' section, which will be made available for you.

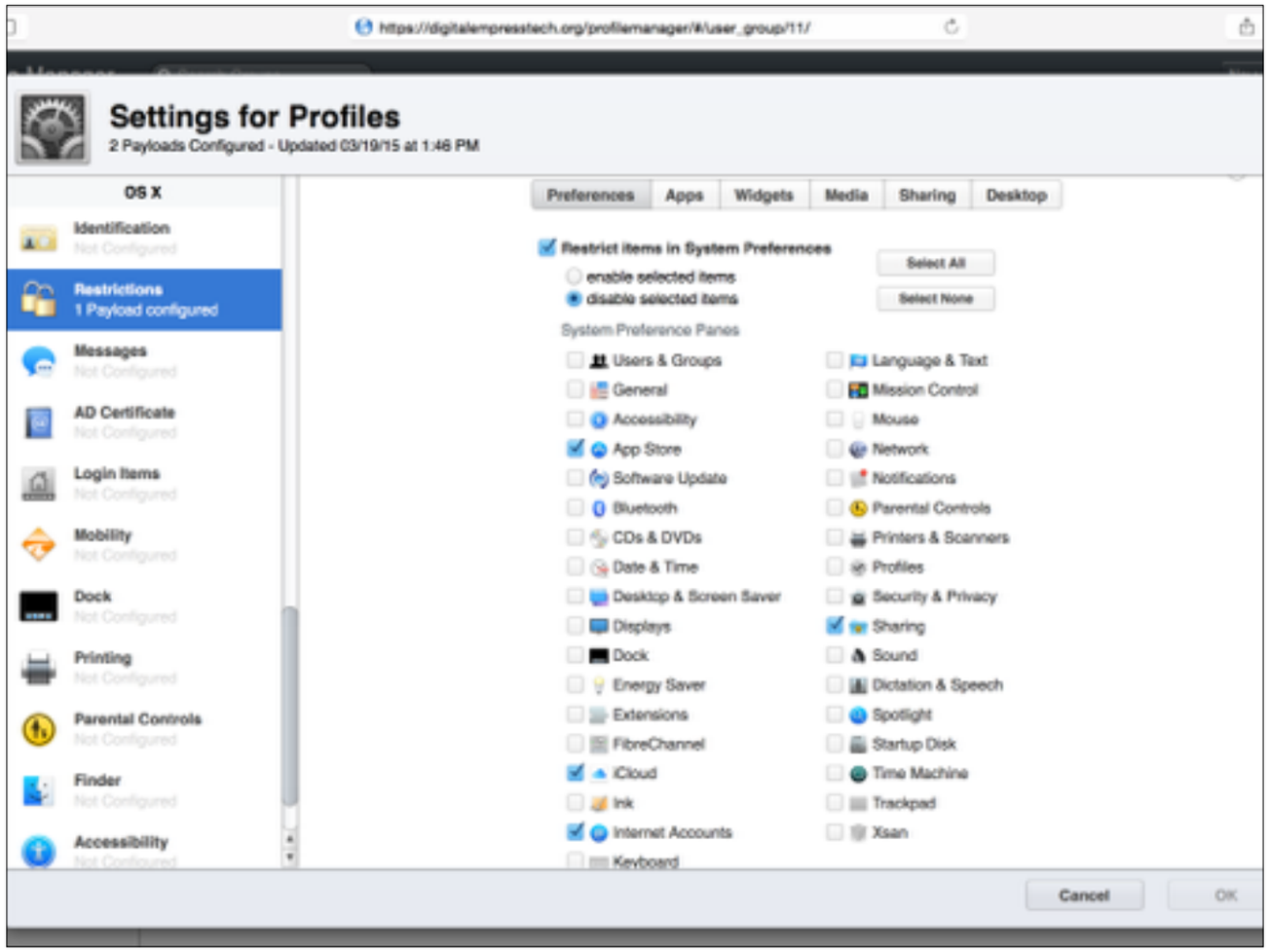
These profiles are compatible with multiple management systems,

- JAMF's Casper Suite and the Profile Manager in Apple's OS X Server provide many pre-defined templates
- allow you to add custom plist entries.

Configuration profiles can be utilized to implement a large number of the items we mentioned earlier.

Examples:
Login Window
- *No Auto Login*
- *No Guest Account*
- *No Fast User Switching*
- *No password hint*

Restrictions
- *Block System Preferences*
- *iCloud*
- *Sharing*
- *App Store*
- *App Store-Updates Only*
- *Admin to Update*
- *CD/USB Mounting*
- *Screen Saver*



Making configuration profiles in profile manager, OS X Server, are pretty simple. Mostly its just click and go.

This screenshot is for restricting items in system preferences. This one profile actual checked off at least 6 requirements for me.

App Store
- blocking the app store helps keep unapproved applications off the system.

iCloud
- blocks the cloud, and iCloud enabled features, like continuity and handoff

Internet accounts
- which would allow a user to setup personal email and iMessage accounts

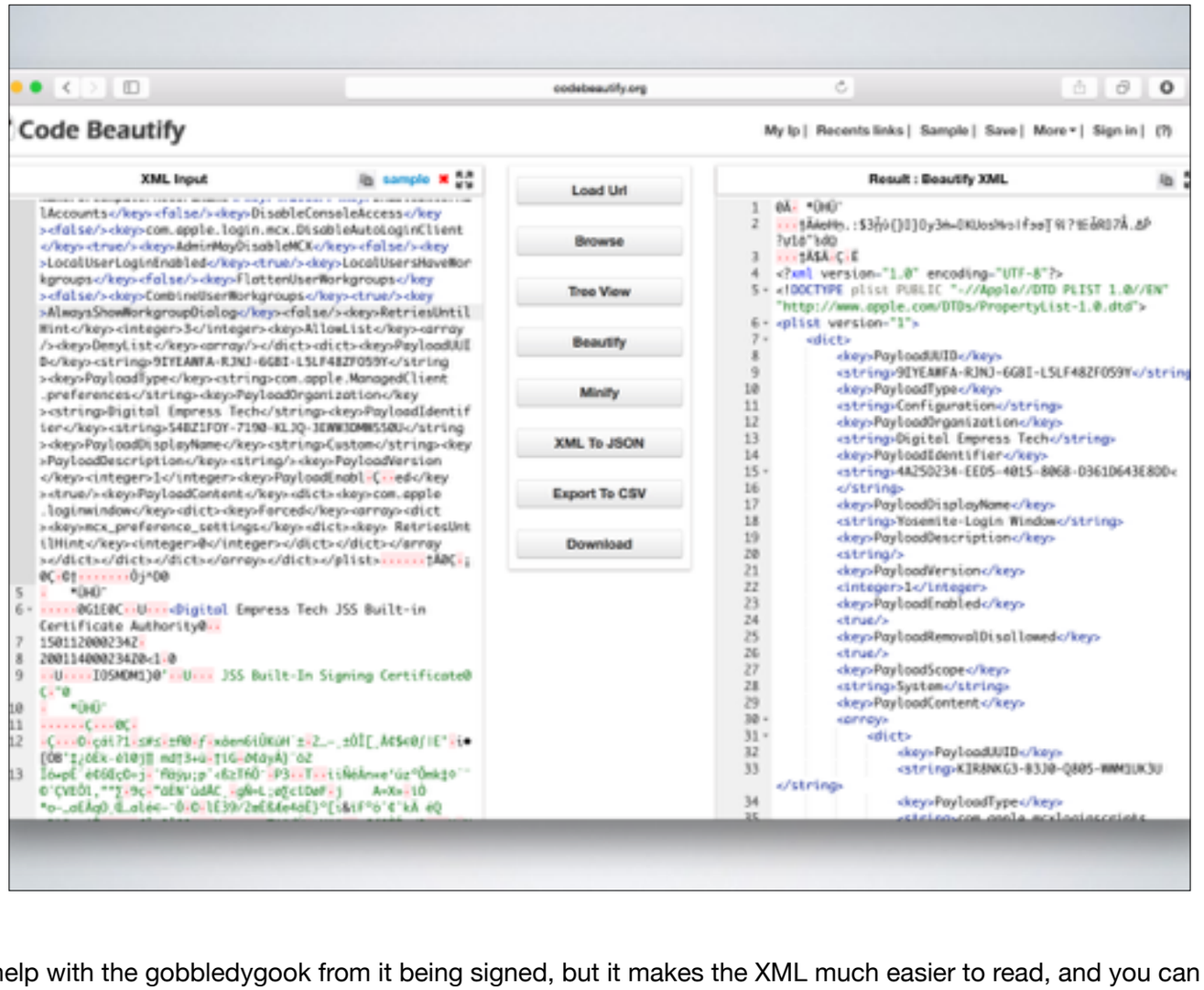
I’ve had success with this method as it blacklists the items you want to block, instead of an old method I used that white listed the ones I wanted to allow. The whitelist method isn’t future proof.

So when I tried to install a Wacom tablet, something with its own preference pane, I couldn’t.

As we learned, a configuration profile is an XML file. This is a sample config profile as is, just opened in xcode or a text editor. They can also be signed, contributing to the difficulty of reading this one.

which is good for your security but bad of your readability.

Now when you create you own, obviously it won't look like this. But if you want to go back and see what a profile you've already made is doing, or if something is failing. Maybe you have to take over from someone with poor documentation skills….regardless, its good to have a way to read them.



Insert codebeautify.org, awesome site, won't help with the gobbledygook from it being signed, but it makes the XML much easier to read, and you can see what your individual keys are actually doing. Very helpful for troubleshooting.

A quick warning, this site is awesome for converting XML from a **fake** example.

However, its still a webpage, so make sure you aren't putting sensitive data into it. Or, try an alternate XML converter.

```
<key>PayloadDisplayName</key>
<string>Login Window</string>
<key>PayloadDescription</key>
<string/>
<key>PayloadVersion</key>
<integer>1</integer>
<key>PayloadEnabled</key>
<true/>
<key>DisableAutoLoginClient</key>
<true/>
<key>AdminHostInfo</key>
<string>HostName</string>
<key>LoginwindowText</key>
<string>Property of Digital Empress Tech</string>
<key>SHOWFULLNAME</key>
<true/>
<key>HideLocalUsers</key>
<false/>
<key>HideMobileAccounts</key>
<false/>
<key>IncludeNetworkUser</key>
<false/>
<key>HideAdminUsers</key>
<false/>
<key>SHOWOTHERUSERS_MANAGED</key>
<true/>
<key>ShutDownDisabled</key>
```

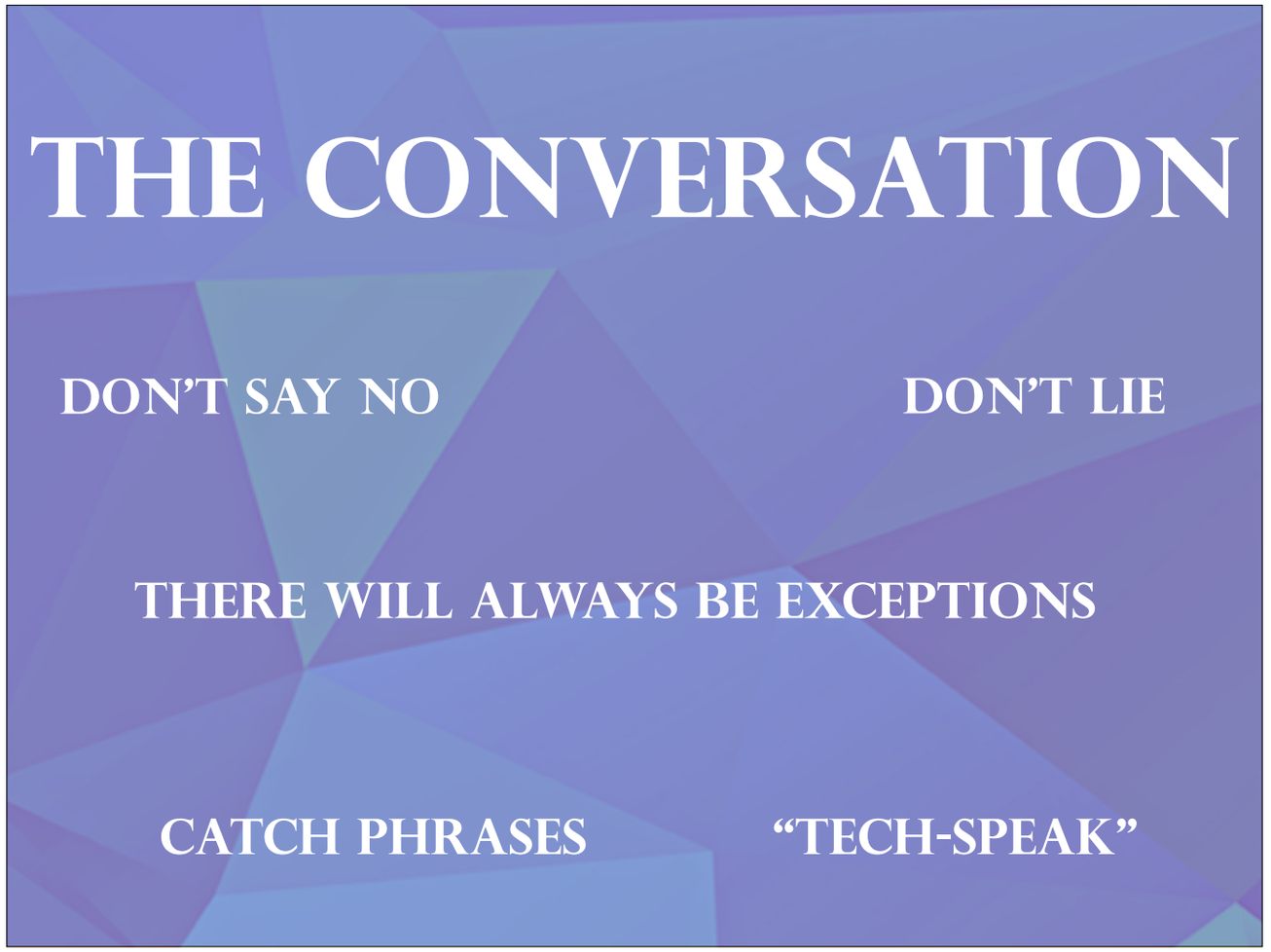

Heres a snapshot of part of my login profile. i have highlighted just a few so you can see what we are doing.
This profile is set to show text on the login window, to show the local users and mobile accounts.

And as there are at least 22 customizable keys in just this one login profile, there is a lot you can change and effect.

PART II
THE USERS

Once you've established what security standards you want to implement, and how to do so, its time to talk to the users.
After all, I can't very well call this presentation "The Joy of Telling Users No" without telling the users no.

Please trust me when I say it is not a good idea to make sweeping changes without talking to anyone about it first. This is not one of those 'life lessons you need to learn on your own' , just don't do it.



Don't say no
- dont start off negative you are here to enforce the policies but you are not a dictator
- there will always be exceptions

I repeat, there will always be exceptions
- don't hold on to anything too tight b/c someone, somewhere, will always have the ability to go over your head; if you don't own the company
- if i hasn't happened to you yet, it will.

Don't Lie
- just admit if you don't know something
- offer to find the answer
- but make sure you follow up, don't leave them hanging
- Once your user loses faith in you, its gone

consider your audience
- language and attitude matter
- regulate your usage of 'tech-speak'

Platitudes, catch phrases and acronyms
- use them sparingly
- you can 'dumb it down' without insulting them *(really don't dumb it down, just use a language they understand)*

All of this combines to give you a positive working relationship with the users.

All of this combines to give you a positive working relationship with the users.

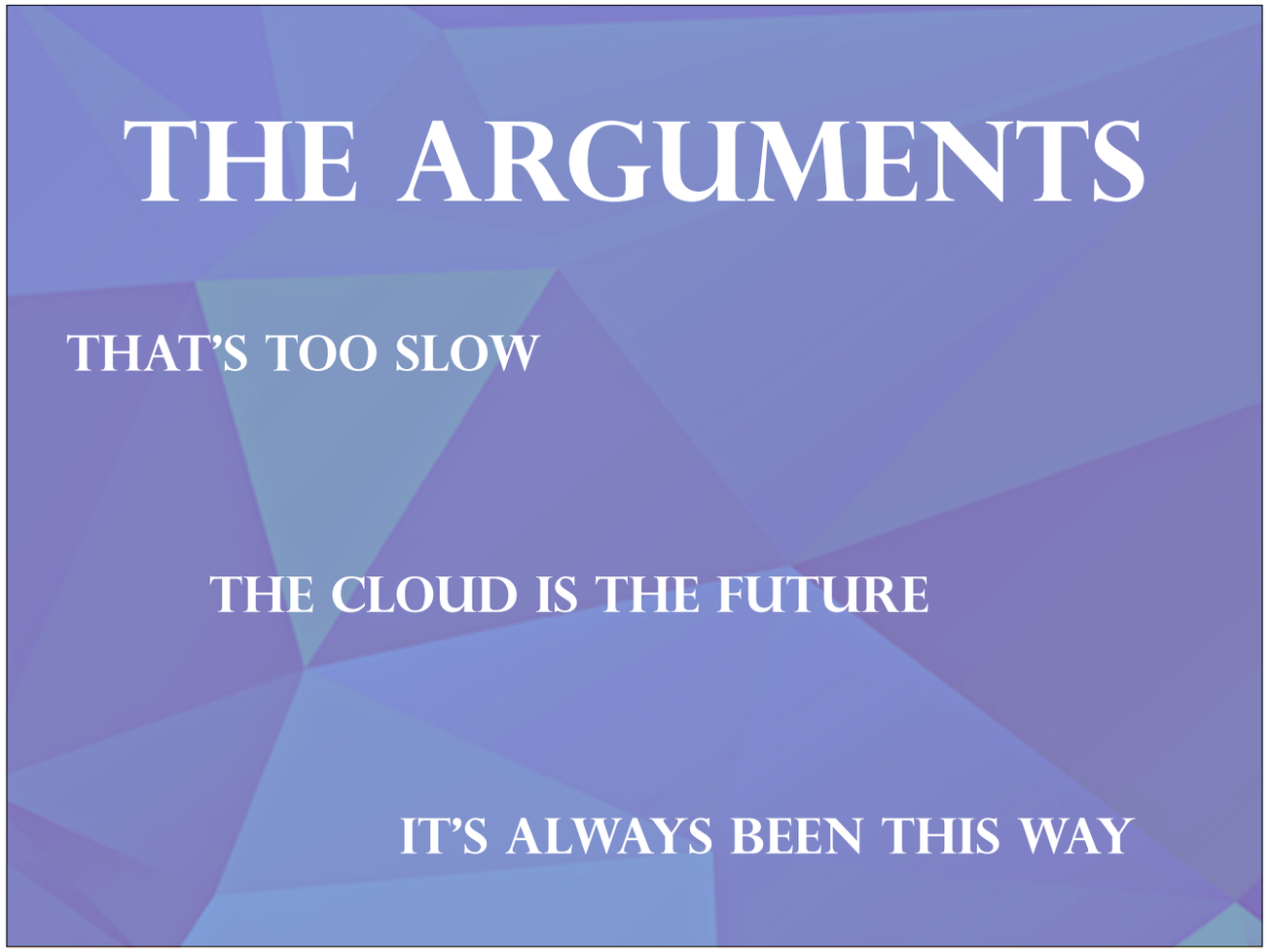
In my experience a lot of the negativity between admins and users is due to a lack of communication.
and an inability to understand each other.

ANTICIPATION, ARGUMENTS & ALTERNATIVES

Anticipation, Arguments and Alternatives.

This part is really for you. Its where you can do a little work in advance, to head off potential problems.

Be prepared to work with your users; and if everything goes right (fingers crossed) you won't have any issues b/c you'll have addressed everything already.



Rarely have a heard an argument with good enough reasons to change your security policy

Its too slow
- ask for specifics.
- Is it really the security policy thats slowing down the machine?

The only thing I've had second thoughts about was Secure empty trash.
- yes its slow, and even slower on filevault,
- using at a minimum one of them.
- And as SSDs keep gaining market share, that argument gets flimsier.

iCloud
- this should be a why, not a why not.
- no good reason, then they don't need it.
- remind them that this is personal information going onto a company machine, would you want all your personal data available for the company to read?
- also, the future? not their decision to make

its always been that way,
- not a valid argument.
- Even 3 year olds know that "because I said so" really isn't a reason.

Offer to facilitate further discussion, if they do present you with a valid reason for making changes, document it, research it and present it to your security and management teams.

Even Moses knows what its like to bring people rules they may not be ‘enthusiastic’ about. ;-)

Remind them:
You are the implementer, not the creator or the enforcer. Security is. (we’re assuming)
 security is the driving force

let them fight the battle

Know what they want before they ask for it, before the argument starts

Provide Applications in advance
- know/understand your environment, different departments will have different needs
- by providing the applications in advance, you users aren't out on the web, downloading from random websites etc

Network Resources, External devices….what the users really want here is access.
Access to Network resources such as shared drives is going to give your user more exposure to sensitive data…..
And access to external devices, like USB drives or CD Burners, might be convenient, but it exposes you and your company to more risk.

Look for a way to lessen this risk as much as possible. If you can't outright ban USB drives, try limiting them to specific machines, and restricting the access those specific machines have to the rest of the company.

Printers, think about how you are going to install printers across your organization.
At first glance you may not think of a printer as a large risk factor, but adding printers needs admin access, and printer Sharing holds a risk of attackers attempting to exploit the print server to gain access to the system.

The Alternatives
- everyone wants to be an admin, but its not necessary for their job.
- And as we've established, admin rights opens the door for a lot more things to go wrong.

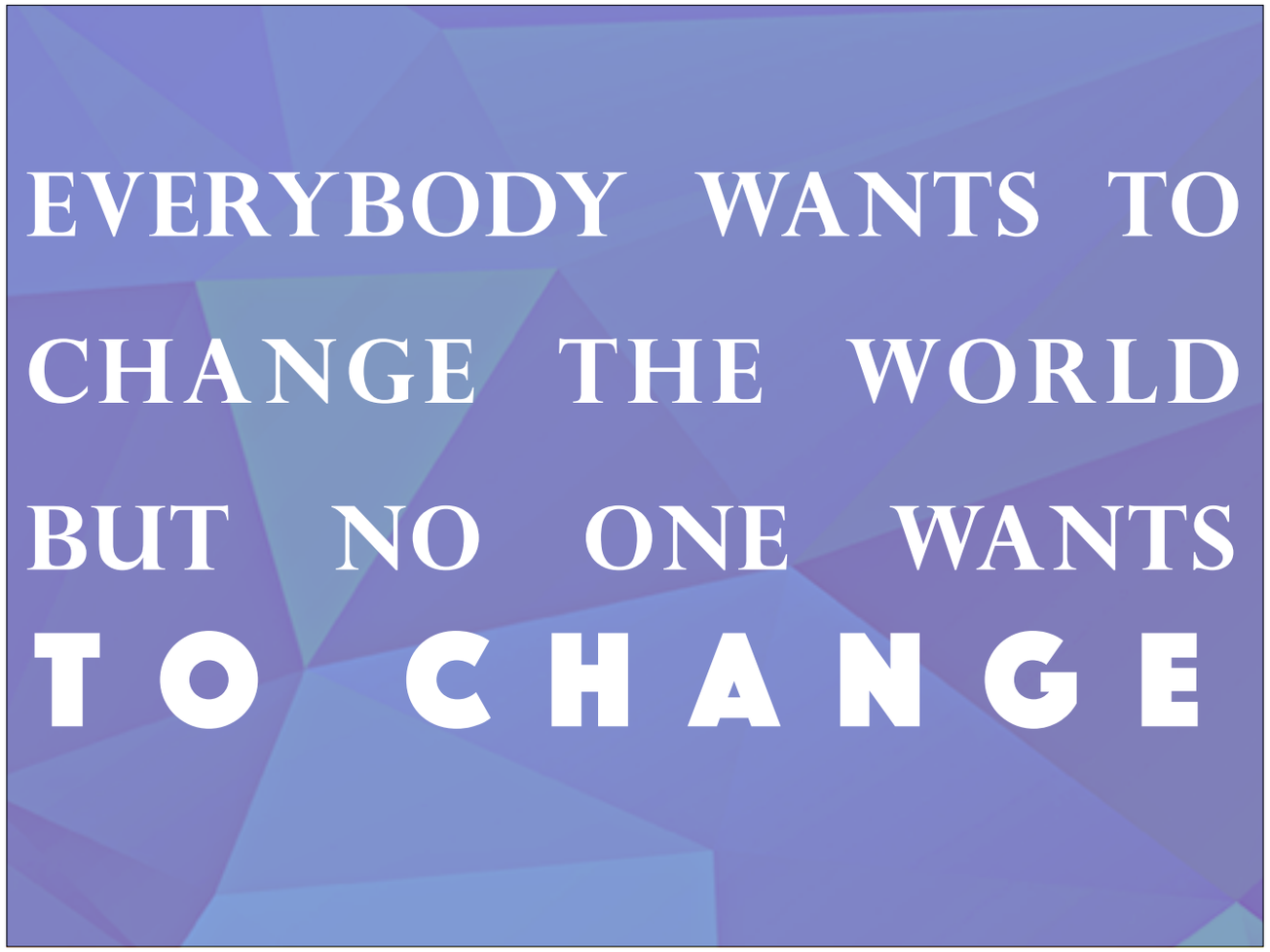
Group Membership additions are an easy alternative.

Developers
- look into adding the users to the 'dev' group
- this will provided the needed access for Xcode without giving full system admin rights

Adding printers
lpadmin group; now a user can add their own printer. Then can also start and stop jobs if a printer messes up *(something that is super annoying to get tickets for).*

Self Service options are a proactive choice. And available for multiple management systems
You can put pre-approved (and pre-tested) applications a self service portal and Admin rights will no longer be necessary to install.

Users can still pick and chose the items they want/need, they maintain a feeling of control and an ability to customize their system, while you maintain security,



Probably the hardest part is to take away something they've already gotten used to. not necessarily b/c the need it but because they are used it it

Additionally, trying to change things one at a time can be frustrating, for you and for the user.

One option for rolling in these changes is to do it all at once.

With apple rolling out a new operating system each year, the OS upgrade is a perfect opportunity to set your new security standards. If users want something, like a new OS, they tend to be more willing to accept the things that come with it.

When users upgrade to the new OS, make it a condition of the upgrade that they must be a standard user.
Explain the changes in security policy on the new OS.
we have seen that its easier to move forward, one change, one adjustment period.

everyone wants change, but no one likes change

Be upfront and honest WHILE clearly showing that what you are changing is absolutely necessary.
Help them see that this is a benefit, a way to protect the company, and them.
ie, with this, the company won't get sued, you won't lose your job.

QUESTIONS?

RESOURCES

CENTER FOR INTERNET SECURITY

https://benchmarks.cisecurity.org/downloads/benchmarks/

CODE BEAUTIFY

http://codebeautify.org/xmlviewer

DIGITAL EMPRESS GITHUB (EXAMPLES)

https://github.com/quovadimus/Digital-Empress-Tech

APPLE REFERENCE: CONFIGURATION PROFILES / PROFILE MANAGER

https://help.apple.com/advancedserveradmin/mac/4.0/#/apdE6F195C7-47EF-48A0-BEE9-0D9B9A24A5A3

http://help.apple.com/profilemanager/mac/2.2/#apd0E2214C6-50F0-48C9-A482-74CEA1D77A9F

TECHWORLD - REMOVING ADMIN RIGHTS WOULD EASE 97 PERCENT OF CRITICAL MICROSOFT FLAWS

http://www.techworld.com/news/security/removing-admin-rights-would-ease-97-percent-of-critical-microsoft-flaws-3605895/

FURTHER READING

Forums:

https://jamfnation.jamfsoftware.com/index.html

Core Dumps:

http://krypted.com/mac-security/core-dumps-in-mac-os-x/

AutoUpdate:

https://derflounder.wordpress.com/2014/12/24/managing-os-xs-automatic-security-updates/

https://derflounder.wordpress.com/2014/12/27/managing-automatic-installation-of-configdata-and-security-software-updates-on-yosemite/

Add standard user to _developer

http://stackoverflow.com/questions/1837889/authorize-a-non-admin-developer-in-xcode-mac-os

Add user to lpadmin

https://macmule.com/2011/07/27/how-to-allow-all-users-to-add-or-remove-printers/

Details on user accounts and groups available at the links above. Samples and terminal commands are available at:
https://github.com/quovadimus/Digital-Empress-Tech