title: "Baldi's Basics Answer Key" date: 2024-01-31 description: "Answer key for Baldi's Basics Image" tags: ["CyberPatriot", "Walkthrough"] type: post weight: 20 showTableOfContents: true

Forensics

Forensics 1

For forensics 1, you had to decrypt the packets in the given pcap file using the ssl keys on the desktop. Through that, you would notice a `mdns poisoning` (2nd answer) attack that began with the url `http://youtube/` (1st answer). Searching through the packets for YouTube video links, you would come across `https://www.youtube.com/watch?v=IFERaX0EwDU` and `https://www.youtube.com/watch?v=f6hmHgenVQg` (3rd answer).

Forensics 2

Searching the system for common auto-run areas, you would find a visual basic script at `C:\Windows\SYSVOL\domain\Policies\{31B2F340-016D-11D2-945F-00C04FB984F9 \MACHINE\Scripts\Startup\startup.vbs` (1st answer), which added the `Administrator` user with the password `Sup3rS3cu4eP@ssw0rd!` (2nd answer).

Forensics 3

This question could have multiple solutions. Some options would be to analyze the image's certificates with a fresh install, check recently installed certificates, or analyze the malware installed by the attacker. Analyzing the dll at `C:\Windows\System32\cbdhsvc_57ffe2.dll` would reveal it was signed by `Microsoft Root Certificate Authority 2012` (1st answer). Checking this certificate would reveal the expiration date to be `12/31/2029 4:00:00 PM` (2nd answer).

Forensics 4

The trickery described in the forensics is that there are two different reset password options when you right-click a user. One of these would work normally, while the other would run a PowerShell script adding back the Null user. This is done through display-name specifiers in the active directory located at `CN=user-Display,CN=409,CN=DisplaySpecifiers,CN=Configuration,DC=baldi,DC=local` (1st answer). The specific attribute in this object that was modified is `adminContextMenu` (2nd answer), and the script it is calling is located at `C:\Users\Administrator\AppData\Roaming\Microsoft\Windows\Templates\reset_password.ps1` (3rd answer).

Vulnerabilites

User Auditing

```
Administrator account is not enabled - e
Removed unauthorized user Null - e
User Arts and Crafters is not a Domain Admin - e
User Bully no longer uses DES encryption for kerberos - e
User Beans no longer has a SID History - h
```

You can read more about SID history here: https://attack.mitre.org/techniques/T1134/005/ (https://attack.mitre.org/techniques/T1134/005/).

Account Policy

```
Passwords must meet complexity requirements - e
```

Local Policy

```
Domain Users can no longer backup files and directories - e
LDAP server requires signing to be negotiated with clients - e
Storage of credentials is no longer allowed for network authentication - e
Strengthened default permissions of internal system objects - e
UAC switches to secure desktop when prompting for elevation - e
```

Defensive Countermeasures

```
Windows Defender blocks process creations originating from PSExec and WMI commands - e
Windows Defender examines DNS queries for exfiltration attempts - m
Windows Defender runs in a sandboxed environment - h
IPsec forces Diffie-Hellman for key exchange - m
IPsec no longer uses MD5 for hashing - m
```

Blocking process creations from PSExec and WMI is the attack surface reduction rule `d1e49aac-8f56-4280-b9ba-993a6d77406c`.

Examining DNS queries for exfiltration is `EnableDnsSinkhole` in MpPreference options.

Sandboxing Windows Defender is the environment variable `MP_FORCE_USE_SANDBOX`.

The IPsec vulnerabilities are just located in the advanced Windows Defender firewall properties.

Uncategorized Operating System Settings

```
IPv6 component disabled - e
Disk quotas disabled for C: - e
PATH environment variable does not contain C:\temp - m
Your SSL keys are no longer logged - m
Everyone no longer has write permissions to the Security event log - m
Non-administrator account are not allowed to make root installations into their own HKCU certificate store - sh
Windows Platform Binary Table (WPBT) functionality disabled - sh
```

IPv6 isn't required and only adds to the system's attack surface.

Disk quotas were set to deny any storage to users, not allowing them to conduct their daily tasks.

The `C:\temp` directory was writeable by everyone while being in the PATH variable, making it vulnerable.

The student's SSL keys are being dumped with the `SSLKEYLOGFILE` environment variable, allowing anyone to record network traffic and later decrypt it.

Everyone could write to the event log, which means they could implant fake logs, clear their tracks, etc.

By default, non-administrator users can make certificate installations into their own HKCU store. This can be leveraged by hackers to install their own malicious certificates, even with non-privileged accounts. There is an undocumented registry key modification outlined by MITRE to mitigate this vulnerability by setting `HKLM\SOFTWARE\Policies\Microsoft\SystemCertificates\Root\ProtectedRoots\Flags` to `1`. (https://attack.mitre.org/techniques/T1553/004/ (https://attack.mitre.org/techniques/T1553/004/))

There are many persistence mechanisms in Windows; however, not all need to be stored in Windows. During the Windows boot, `smss.exe` calls the `NtQuerySystemInformation()` function with the deprecated parameter `0x85`. This scans UEFI tables for WPBT (Windows Platform Binary Table) and passes them back to `smss.exe`. `smss.exe` then stores this piece of UEFI memory into `wpbbin.exe`, takes a command line from the same parameter, checks `wpbbin.exe` integrity with `IMAGE_DLLCHARACTERISTICS_FORCE_INTEGRITY`, and `wpbbin.exe` is executed. While extremely complicated, this means that attackers could execute code (written using only `ntdll.dll` and no `Win32Api` calls) on boot. With this, attackers can obtain UEFI-level persistence and inject executables into Bitlocker-encrypted drives. The WPBT functionality can be completely disabled, thus mitigating any attack vectors, by setting the undocumented registry key `HKLM\SYSTEM\CurrentControlSet\Control\Session Manager\DisableWpbtExecution` to `1`.

Service Auditing

```
Windows Defender Firewall startup type is automatic and running - e
Everyone no longer has full control over Windows Event Log service - sh
Windows Defender Service runs in Protected Process Light (PPL) mode - sh
```

Permission doesn't only apply to basic file structures. Services have their own set of permissions stored in their SDDL string. The Event Log Service's SDDL has been modified to allow everyone to have full control over the service. This means that anyone could start or stop the service, as well as a lot more malicious actions.

Many services in Windows run in PPL (Protected Process Light) mode, protecting them from undue modifications. The most commonly remembered one is LSASS; however, Windows Defender is also one such service. On the image, Windows Defender is no longer running in PPL mode.

Operating System Updates

```
Windows automatic updates configured - e
```

Application Updates

```
Firefox updated - e
```

Prohibited Files

```
Removed startup script with plaintext administrator password - e
Removed unauthorized partition with prohibited files - m
```

Unwanted Software

```
Removed Powershell 2.0 - m
Removed DSInternals powershell module - h
```

Malware

```
Removed malicious Microsoft Root Certificate Authority 2012 certificate - m
Removed malicious display specifier imitating reset password option - m
Removed service running skeleton key persistence mechanism - m
```

Application Security Settings

```
LDAP no longer allows anonymous calls to the NSPI RPC bind method - m
Authenticated users can no longer edit the defualt domain policy GPO - h
Authenticated users may not join computers to the domain - h
DNS version query disabled - e
DNS server no longer performs recursive name resolution - e
baldi.local zone only allows secure dynamic updates - e
DHCP server is authorized - e
DHCP audit logging enabled - e
DHCP IPv4 scope uses name protection - e
Enabled Restricted admin for RDP - e
RDP Shadowing blocked at firewall level - h
```

By default, authenticated users may join up to 10 computers in the domain, which provides full control to the user. This has been previously used in `CVE-2021-42287` to escalate privileges. Thus, it's recommended to set the `ms-DS-MachineAccountQuota` attribute to 0.

There was a firewall rule that specifically allowed RDP shadowing. It is best to block RDP shadowing at the firewall level to provide an extra level of security.