University of New Haven

University of New Haven
Digital Commons @ New Haven

Electrical & Computer Engineering and Computer Science Faculty Publications

Electrical & Computer Engineering and Computer Science

1-6-2018

An Overview of the Usage of Default Passwords

Brandon Knierem
University of New Haven

Xiaolu Zhang
University of New Haven

Philip Levine
University of New Haven

Frank Breitinger
University of New Haven, frank.breitinger@unil.ch

Ibrahim Baggili
University of New Haven, ibaggili@newhaven.edu

Follow this and additional works at: https://digitalcommons.newhaven.edu/electricalcomputerengineering-facpubs

 Part of the Computer Engineering Commons, Electrical and Computer Engineering Commons, Forensic Science and Technology Commons, and the Information Security Commons

Publisher Citation

Knieriem B., Zhang X., Levine P., Breitinger F., Baggili I. (2018) An Overview of the Usage of Default Passwords. In: Matoušek P., Schmiedecker M. (eds) Digital Forensics and Cyber Crime. ICDF2C 2017. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol 216, pp. 195-203. Springer, Cham.

Comments

This is the authors' accepted version of the paper published in *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (LNICST).* The volume encompasses the proceedings of the 9th EAI International Conference on Digital Forensics & Cyber Crime, October 9-11 in Prague. The version of record for the proceedings volume may be purchased from the Springer web site.
The authors' extended version of the paper is attached below as a supplementary file.
Dr. Baggili was appointed to the University of New Haven's Elder Family Endowed Chair in 2015.

An Overview of the Usage of Default Passwords

Brandon Knieriem, Xiaolu Zhang, Philip Levine, Frank Breitinger, and Ibrahim Baggili

Cyber Forensics Research and Education Group (UNHcFREG)
Tagliatela College of Engineering University of New Haven, West Haven CT, 06516, United States
{bknie1, plevi1@unh.newhaven.edu},{XZhang, FBreitinger, IBaggili@newhaven.edu}

Summary. The recent Mirai botnet attack demonstrated the danger of using default passwords and showed it is still a major problem. In this study we investigated several common applications and their password policies. Specifically, we analyzed if these applications: (1) have default passwords or (2) allow the user to set a weak password (i.e., they do not properly enforce a password policy). Our study shows that default passwords are still a significant problem: 61% of applications inspected initially used a default or blank password. When changing the password, 58% allowed a blank password, 35% allowed a weak password of 1 character.

Key words: Default passwords, applications, usage, security

1 Introduction

In October 2016, a large section of the Internet came under attack. This attack was perpetuated by approximately 100,000 Internet of Things (IoT) appliances, refrigerators, and microwaves which were compromised and formed the Mirai botnet. Targets of this attack included *Twitter*, *reddit* and *The New York Times* all of which shut down for hours. The Mirai botnet was created by abusing default credentials in IoT devices [7, 14]. Besides devices, there are also applications permitting users access to critical central resources such as Database Management Systems (DBMS), Web Server Applications, and Content Management Systems (CMS). For instance, in July 2014 hackers attacked HealthCare.gov [18]. Fifteen days later HealthCare.gov released a statement that only the test servers were hacked and no personal information was compromised. The attack occurred because the manufacturer's default password on the server had not been changed. Days later, despite reporting on this vulnerability, the default password had still not been updated [2].

These findings motivated us to perform two short surveys with the goal to start a discussion in the field about the usage of develop passwords: The first was to examine applications such as DBMS, Web Server Applications, and CMSs that enable a default password during initial configuration. Results show the most applications have default credentials. The second survey was conducted on developers to understand the use of default passwords. The results indicate

that many services are designed with default passwords to bypass authentication to provide immediate, temporary access for quick, convenient initial set up of infrastructure and should only be used during this installation phase.

Remark: The extended version of this article named 'An Overview of the Usage of Default Passwords (extended version)' can be accessed through digital commons[1].

2 Literature review

"Passwords are an ubiquitous and critical component of many security systems" [19]. Therefore, it is important to create secure passwords that are difficult to compromise. For instance, according to [9], a strong password policy requires a minimum number of characters, different types of characters, and specify how frequently users should change their passwords. More recently, National Institutes of Standards and Technology (NIST) Digital Authentication Guidelines have suggestions for improving security and reduce common issues [6]. These suggestions can be broken down into several main categories: hashing passwords, increasing user friendliness, enforcing an 8-character minimum, and banning common passwords. It suggests avoiding password rules, password hints, security questions, and never forcing arbitrary password changes [22]. These suggestions are a step in the right direction for new password policy, though despite being nearly a year old, have yet to be implemented by many application developers.

2.1 Breaches Exploiting Default User Credentials

Although guidelines and warnings regarding default passwords exist, there are still many incidents involving default credentials. According to [4], "very few open source vendor advisories have mentioned default passwords, whereas they appear with some regularity in closed source advisories, even in the top 10 [vulnerabilities] as recently as 2005".

A newer study named the Verizon Data Breach Report examined 621 corporate breaches. "The analysis found that 78% of initial intrusions into corporate networks were elementary. Many attackers use a phishing attack, convincing employees to give up credentials, or brute force attack, taking advantage of weak or default passwords on remote services to gain initial access to the network" [20]. Unfortunately, the report did not mention, out of 78% how many constituted weak passwords or default passwords. Notwithstanding, some of the recent breaches that were attributed to the misuse of default passwords. Utah Department of Health] suffered a breach of 780k Medicaid patient health records [12] in addition to compromising more than 255,000 social security numbers [17]. Attackers achieved complete access to the system using a default password. A Bank of Montreal's ATM was hacked by two 14 year old children; they used the machine's default password [11]. Emergency Alert System (EAS)] equipment

[1] http://digitalcommons.newhaven.edu

used to broadcast warnings was hacked by exploiting default passwords. After the breach, the hackers sent out an alert warning the public of a 'zombie attack.' [11]. Electronic Highway Billboards were attacked in June 2014. The hacker changed the signs to display their Twitter/hacker handle for all the highway drivers to see. This was an act of mischief reported by [8].

A recent WordPress incident demonstrated that the usage of a default username can result in a tremendous security risk. In case of WordPress, the default username is always 'admin'. Hackers used that knowledge and used a botnet to brute force 90,000 IP addresses hosting different software [1]. Unfortunately, the report did not release how successful this attack was.

2.2 Taking advantage of default passwords - tools, scripts and malware

Attackers, often taking an opportunistic approach, realized the potential in abusing default passwords to access a system. Thus, there are several tools, scripts, and malware that can be used for this purpose. Work [16] states "the most common password lists found using Internet search engines were default password lists. These lists contain passwords used by hardware manufacturers as the default security setting". In a recent article, [3] mentioned several tools that focus on exploiting default passwords. For instance, Cisco OCS Perl Script scans Cisco devices on a network by inputting 'cisco' into the password form. Metasploit includes multiple modules used for network default password scanning.

On the other hand, several worms exist that use default passwords to propagate. According to [13],the 'Voyager Alpha Force' worm was used to demonstrate a vulnerability on Microsoft's SQL Server with an administrator blank password using the default port: 1433. Similarly, MySQL required no password at the time of installation. A worm named "MySpooler" infected 8000 hosts at a rate of 100 hosts per hour [10]. In 2005, an anonymous developer disclosed a proof-of-concept worm that targeted Oracle databases using default usernames and passwords [23]. A particularly malicious worm implementation uses blending viruses; which are viruses that run a daily Internet scan for vulnerabilities. One of the main functions of them are to find well known default passwords [5].

Work [15] triggered malnets; a combination of malware and bots initiated malware attacks on routers. A similar experiment was also performed by [24]. Regarding wireless malware propagation: 16.7% of routers were set to be configured with default settings. Only 10% of these routers used default passwords or did not have passwords set.

3 Applications analysis

To analyze the impact of default passwords we examined database management systems, Web server applications, and content management systems. We decided to focus on web applications as they are easily accessible and cater to a broad

audience. More precisely, we investigated Relational Database Management Systems (RDBMS), Web Server Applications (WSA), and Content Management Systems (CMS). After locating a comprehensive list containing the major applications in all three categories, our methodology is as 1) For each identified application, search for documentation and identify the default credentials / settings. 2) Download and install a free or evaluation version of each application. Prioritize installation on Windows 10 (64-bit), then Ubuntu Linux 16.04.2, and finally Mac OS Sierra 10.12.5. Use default configurations and procedure; do not use advanced or customized installation options. 3) If a default database is not created during installation, create one immediately after installation. 4) Note any prompts, or lack thereof, regarding security policy enforcement. 5) Assign each conclusive application a password policy quality value on a scale of 0 to 4. This was loosely based on an IBM's classification [21].

3.1 Results

In total, $n = 90$ applications were analyzed where 62 applications yielded conclusive results and 28[2] had inconclusive results due to licensing restrictions. An overview of the results is given in Tables 1 and 2. Of the 62 conclusive applications, 41 applications had commercial licenses and 21 were open source. To analyze the applications, 51 applications were installed on Windows 10 (64-bit), 8 were installed on Linux-x86, four were web services, and one was installed on Mac OS. Note, two applications were a pre-release version (0.1 - 0.9/Alpha/-Beta), the remaining 60 applications were a release version (1.0+) (97%).

In total, 30 applications featured a default user name, the most frequent were "Admin" or "root". 6 (10%) applications featured a default password. 32 (52%) applications featured a default blank password for the default user account. All applications featuring a default password also featured a default user name.

Lastly, we analyzed the quality of the passwords according the IBM classification [21]. Overall, 36 (58%) applications were categorized as having a level 0 policy, 22 (35%) applications were categorized as having a level 1 policy. Two applications were categorized as having a level 2 policy. One application was categorized as having a level 3 policy. Finally, only one application that met the requirements for a level 4 policy, which is interesting as this is what most modern online portals require.

4 Qualitative survey of default credential use

This section tries to understand why default user credentials / passwords are still so widely used. Therefore, we created a question for software developers, computer engineers, and security experts: why many applications still come with a

[2] Actian Ingres, Actian Vector, CA Datacom, CA IDMS, Clarion, Clustrix, Empress Embedded Database, EXASolution, eXtremeDB, GroveSite, IBM PureSystems, Infobright, Linter, Microsoft Visual FoxPro, NexusDB V4 Windows, NonStop SQL, Openbase, Postgres Plus Advanced Server, R:Base, SAP ADS, SAP Anywhere, SAP HANA, SAP Sybase ASE, SAP Sybase IQ, SQL Azure, SQream DB, UniData, Vertica

default user name password and do not require the user to set new credentials according to a reliable password policy? The question was distributed online in 20 software developer forums, advertised to 30 groups on Quora, and other forums. The question was also sent directly to 35 users on Quora who are known developers and 10 professors from the University of New Haven and the University of Bridgeport (IRB approval was obtained prior to the start). The question received high exposure; in one instance over 2,800 individuals accessed or viewed the question on Quora. However, the response rate was low. In total, we only received 20 responses. 6 users blamed the developers for writing a sloppy code. A Web Development project manager on Quora described a situation: "I ran across a custom WordPress / Yii app that used the same password by default. As the dev manager, I pointed out that this was a major flaw. Got told that it was but wasn't urgent. Until a hack happened..."The CEO of mid-size online company on LinkedIn explained a situation where a default password is used: "I need to install my Lazarus application on 20 clients. Can you imaging running through the setup process with password policies right from the start? Do you see how much more time you'll need to spend? ... I imagine you know the hassle of dealing with OS permissions, DB permissions (different user), application permissions, and then user roles. Yes, it is possible to have a security policy in place from the start, but do you see how much more difficult it gets?"

5 Discussion & conclusion

Applications are designed to provide the best user experience to their customers and reduce setup time. Especially when the administrator needs to install the application on multiple devices in succession. The default passwords in this study demonstrate this by being easy to remember and utilize for multiple devices. For instance, most of applications used 'password', 'admin', 'dba' etc as default passwords.

Many of these applications accepted a single character as a valid user name or password. A user may choose a more complex password, but because there is often no requirement for special characters or total character count, the user may choose the easiest, most convenient credential solution.

In summary, this article surveyed a well-known default password issue on 21 open-sourced applications and 41 commercial applications. Out of the 62 applications, we found that 32 applications featured a default user name, 6 applications featured a default password and 32 applications accepted empty passwords. In total, 38 applications surveyed can lead an administrator using default user credentials. Meanwhile, in order to evaluate the password policy we also scored the applications with IBM password quality scale. 36 of applications scored with '0', having no password policy. 22 of applications scored a '1', meaning that a single character password is acceptable, the weakest possible password policy. Only 4 applications had an acceptable password policy. To explain why practitioners may keep default user credentials of the DBMS on their own database system,

we distributed a survey on Quora and responded by variety roles such as web developer, system manager, CEO etc (Sec. 4).

Acknowledgements. Special thanks go to Mohammed Nasir who initially started this research project and Matthew Vastarelli for supporting us.

References

1. Logan Booker. Brute Force Attack Targets WordPress Sites With Default Admin Username, 2013.
2. Rebecca Carroll. Breached healthcare.gov server still had default password, 2014.
3. Brad Casey. Network security risks: The trouble with default passwords, 2014.
4. Steve Christey and Robert A Martin. Vulnerability type distributions in cve. *Mitre report, May*, 2007.
5. John Gordineer. Blended threats: A new era in anti-virus protection. *Information Systems Security*, 12(3):45–47, 2003.
6. Garcia Grassi. Digital identity guidelines. *National Institute of Standards and Technology*, 2016.
7. Nyman Hypponen. The internet of (vulnerable) things: On hypponen's law, security engineering, and iot legislation. *Technology Innovation Management Review*, 7(4):5–11, Apr 2017.
8. KrebsonSecurity.com. They hack because they can, 2014.
9. Flavio Martins. Creating strong password policy best practices, 2014.
10. Stephen Northcutt. The risk of default passwords, 2007.
11. Thu Pham. Default passwords: Breaching atms, highway signs & pos devices, 2014.
12. Duo Security. Utah department of health (udoh) breach, 2012.
13. Microsoft Customer Support. An unsecured sql server server that has a blank (null) system administrator password allows vulnerability to a worm, 2005.
14. Symantec Security Response. Mirai: what you need to know about the botnet behind recent major ddos attacks, Oct 2016.
15. Patrick Traynor, Kevin Butler, William Enck, Patrick McDaniel, and Kevin Borders. malnets: large-scale malicious networks via compromised wireless access points. *Security and Communication Networks*, 3(2-3):102–113, 2010.
16. RP Van Heerden and JS Vorster. Statistical analysis of large passwords lists, used to optimize brute force attacks. 2009.
17. JaiKumar Vijayan. Weak passwords still the downfall of enterprise security, 2012.
18. Kate Vinton. Data breach bulletin: Home depot, healthcare.gov, jp morgan, 2014.
19. Kim-Phuong L Vu, Robert W Proctor, Abhilasha Bhargav-Spantzel, Bik-Lam Belin Tai, Joshua Cook, and E Eugene Schultz. Improving password security and memorability to protect personal and organizational information. *International Journal of Human-Computer Studies*, 65(8):744–757, 2007.
20. Robert Westervelt. Verizon data breach report finds employees at core of most attacks, 2013.
21. Christie Williams and Katherine Spanbauer. Understanding password quality, 2001.
22. Wisniewski. *Naked Security*, 2016.
23. Joshua Wright. Oracle worm proof-of-concept, 2005.
24. Stefano Zanero. Wireless malware propagation: A reality check. *Security & Privacy, IEEE*, 7(5):70–74, 2009.

Table 1. Surveyed Applications

Name	Version/ Release	Platform	Commercial/ Open-Source	License	Default Username	Default Password	Password Policy Quality
4th Dimension	16.1	Windows	Commercial	30-Day Evaluation	"Administrator"	None	0
Adabas	2016 April	Windows	Commercial	Community Edition	Inherits user account.	Inherits user password.	0
Alpha Five	V12	Windows	Commercial	30-Day Evaluation	"Admin"	None	0
Altibase	6.5	Linux-x86	Commercial	Community Edition	None	None	0
Amazon Aurora	N/A	Web-Service	Commercial	N/A	None	None	2
Apache Derby	10.13.1.1	Windows	Open-Source	N/A	None	None	0
Apache OpenOffice.org Base	4.1.3	Windows	Open-Source	N/A	N/A	N/A	0
Apache Trafodion	2.1.0	Windows	Open-Source	N/A	None	None	1[2]
Base X	8.6	Windows	Open-Source	Free Version	"admin"	"admin"	0
ClickHouse	1.1.54189	Linux-x86	Open-Source	N/A	None	None	0
CSQL	3.3	Linux-x86	Open-Source	N/A	None	None	0
CUBRID	10.0.0.1376	Windows	Open-Source	N/A	"admin"	"admin"	2[3]
Database Management Library (C++)	1.0	Windows	Open-Source	N/A	None	None	0
DataEase	6.5 Demo	Windows	Commercial	N/A	"labadmin"	None	0
Dataphor	3.1.6143	Windows	Open-Source	N/A	"admin"	None	0
dBase PLUS	11.2	Windows	Commercial	30-Day Evaluation	None	None	0
Drupal	8.3.2	Windows	Commercial	Free Version	None	None	1
EnterpriseDB	9.6	Windows	Commercial	Standard Version	"postgresql"	None	1
FileMaker Pro	15	Windows	Commercial	Trial Version	"Admin"	None	0
Firebird	3.0.2	Windows	Open-Source	N/A	N/A	N/A	1
FrontBase	8.28	Windows	Commercial	Free Version	None	None	0
Google Fusion Tables	N/A	Web Service	Commercial	Free Version	Google Account	Google Account	3[4]
Greenplum	5.0.0-alpha.3	Linux-x86	Open-Source	N/A	None	None	0
H2	1.4.195	Windows	Open-Source	N/A	"sa"	None	0
Helix	7.0.2	Mac OS	Commercial	Demo Version	None	None	0
HSQL	2.4.0	Windows	Open-Source	N/A	"SA"	None	0
IBM DB2	11.1	Windows	Commercial	Trial Version	"db2admin"	None	1
IBM DB2 Express-C	11.1	Windows	Commercial	Trial Version	"db2admin"	None	1
Informix Enterprise	12.10	Windows	Commercial	Time-Limited	"informix", "ifxjson"	None	0
InterBase	2017	Windows	Commercial	Trial Version	"SYSDBA"	N/A	1

0: No password policy.
1: Password policy only requires a single character.
2: Requires a minimum number of characters but can be compromised without a computer.
3: Requires a minimum number of characters but can still likely be compromised with a computer.
4: Requires a minimum number of characters, numbers, and special characters, and would be difficult to compromise.
[1]: Fully custom credentials required.
[2]: Forces custom credentials following login with defaults.
[3]: Two-factor authentication required.

Table 2. Surveyed Applications (Continued)

Name	Version/ Release	Platform	Commercial/ Open-Source	License	Default Username	Default Password	Password Policy Quality
InterSystems CachÃ©	2017.1	Windows	Commercial	Evaluation Version	"_SYSTEM", "Admin", "SuperUser", "forensics", "CSPSystem"	N/A	1[2]
JBoss Web Console	6	Windows	Commercial	Free Version	"Admin"	"Admin"	0
Joomla	3.7	Windows	Commercial	Free Version	"admin"	None	1
LibreOffice Base	5.3.3	Windows	Open-Source	N/A	None	None	0
MariaDB	10.3	Windows	Open-Source	Free Version	"root"	N/A	1
Microsoft Access	16.0	Windows	Commercial	Office 2016	None	None	0
Microsoft SQL Server	2016 SP1	Windows	Commercial	Express Edition	"sa"	None	0
Mimer SQL	10.1	Windows	Commercial	Trial Version	"SYSADM"	N/A	1
MonetDB	11.25.21	Windows	Open-Source	Free Version	None	None	0
mSQL		Linux-x86	Commercial	Free Version	"root"	None	0
MySQL	5.7.18.1	Windows	Commercial	Community Edition	"root"	None	0
neo4j	3.2	Windows	Commercial	Evaluation	"neo4j"	None	1[1]
NexusDB	V4	Windows	Commercial	Server Trial Version	N/A	N/A	1
NuoDB Database	2.6.1	Windows	Commercial	Community Edition	"dba"	"goalie"	1[1]
NuoDB Domain		Web Service	Commercial	Community Edition	None	None	1
OpenLink Virtuoso	6.0	Windows	Commerical	Trial Version	N/A	N/A	1
Oracle RDBMS	7.3	Windows	Commerical	Free Version	N/A	N/A	0
Oracle TimesTen		Windows	Commercial	Free Version	N/A	N/A	1[2]
Orange HRM	3.3.1	Windows	Open-Source	N/A	None	None	1[2]
Polyhedra	8.6.1	Windows	Commercial	Lite Version	None	None	0
PostgreSQL	9.6	Windows	Open-Source	N/A	"postgres"	None	0
RDM Server	8.4	Windows	Commercial	Trial Version	N/A	N/A	1[2]
SAND CDBMS	8.1	Windows	Commercial	Free Version	"DBA"	None	0
SAP MaxDB	7.8.02.39	Windows	Commercial	Free	"DBADMIN"	N/A	1
ScimoreDB	4.0	Windows	Commercial	Freeware	None	None	0
SQLBase	12.0	Windows	Commercial	Trial Version	"SERVER1"	"SECRET"	0
SQLite	3.18	Windows	Open-Source	N/A	None	None	0
Tableau (Local)	10.2.2 64-bit	Windows	Commercial	14-Day Evaluation	N/A	N/A	0
Tableau (Online)	10.2.2 64-bit	Windows	Commercial	14-Day Evaluation	N/A	N/A	4
Tibero	6.0	Windows	Commercial	30-Day Evaluation	"root", "sys", "syscat", "sysgis", "outln", "tibero", "tibero1"	"tibero", "tibero", "syscat", "sysgis", "outln", "tmax", "tmax"	1
txtSQL	3.0.0b	Windows	Open-Source	N/A	"root"	None	0
Wordpress	4.7.4	Web Service	Open-Source	N/A	None	None	1

0: No password policy.
1: Password policy only requires a single character.
2: Requires a minimum number of characters but can be compromised without a computer.
3: Requires a minimum number of characters but can still likely be compromised with a computer.
4: Requires a minimum number of characters, numbers, and special characters, and would be difficult to compromise.
[1]: Fully custom credentials required.
[2]: Forces custom credentials following login with defaults.
[3]: Two-factor authentication required.