Developing Backup Security Option for User Identity Management

R.Kesavvel[1] N.Balamurugan[2] MR.M.Ramkumar[3]
[1,2]U.G Student [3]Assistant Professor
[1,2,3]saveetha school of Engineering

Abstract— Most cell phones use a password, PIN, or visual pattern to secure the phone. With these types of security methods being used, there is much vulnerability.Biometric security systems have been researched for many years. Some mobile manufacturers have implemented fingerprint scanners into their phones, such as apple phones Motorola atrix here we are trying to implement an efficient security system with agood backup system. To enhance identity acquisition procedures in smart phones and make the process transparent to the user, a novel User Identity. Sensing approach leveraging the unified fingerprint enabled touch panel that combines multiple capacitive TFT based fingerprint sensors directly with the touch screen panel of the Smartphone is proposed and back up user login feature are to be implemented for user interface.

Key words: Backup Security, Android, DVM, UNIX.

I. INTRODUCTION

The traditional approach to ocean monitoring is to deploy oceanographic sensors record the data, and recover the instruments. This approach spends lots of time receiving the recorded information. In addition, if a failure occurs before recovery, all the data would be lost. The ideal solution is to establish real-time communication between the underwater instruments and a control center within a network configuration. Underwater sensor networks can be used in oceanographic data collection, pollution monitoring, disaster prevention, assisted navigation applications. The available bandwidth of the underwater acoustic channel is limited and also dependent on both range and frequency. The sensors are battery power edso power efficiency is a critical issue for under water sensor networks as well.

Extremely long delay in the underwater acoustic channel could lead to collapse of traditional terrestrial routing protocols because of limited response waiting time. According to above facts, designing a suitable network routing protocol in underwater environment is urgent. Sensor nodes with wireless communication can be deployed under the sea level. The sensors detect and then transfer the data from the bottom level to the top level. This paper uses the pressure as a significant indicator for a sensor node to judge its own level of depth. This design reduces great amount of broadcasting hello messages and hence decrease total energy consumption. Based on previous research, transmission range at 150m would be much more efficient for energy concern. The design of our routing mechanism, EUROP, also considers this unique characteristic. EUROP has been implemented and evaluated using NS-2 simulator successfully.

II. INTRODUCTION

Fingerprint and user login security implementations are believed to prevent intrusions and theft against mobile cellular devices. Essentially, a biometric system is used for identification or verification based on physiological and biological factors. Generally speaking, criminal acts are motivated by various reasons. A victim can either be deprived of their cell phone by some form of theft, or be vulnerable to losing sensitive information through a breach in security. More cell phones are being stolen every day because there is a market which demands the supply; some refer to this as a black market which establishes an incentive for theft.

A. Design:

of a novel mobile identity sensing approach that Integrates transparent fingerprint sensors with touch panel for opportunistic identity detection from natural user mobile Device touch interactions.

B. Identification and proposition:

of solutions to tackle some of the must be addressed challenges for implementing the proposed opportunistic identity sensing technology such As optimal placement of fingerprint sensors.

C. Insights gathered:

from the experiments and detailed simulation

Studies that provide valuable guidance and requirements For the hardware design and implementation.

III. SECURITY AND PRIVACY IN INFORMATION SYSTEMS

Smartphones need to be secured from security and privacy violations. A smartphone

Is an example of an information system, it can be analyzed from the point of view

Of information systems security and privacy. In this chapter, we look at key security

Terms, principals and models that can help us start the security analysis of smartphones.

A. Android:

Linux system interacts with the phone hardware and an off-processor cellular radio. The Binder middleware and application API runs on top of Linux. To simplify, an application's only interface to the phone is through these APIs. Each application is executed within a Dalvik Virtual Machine (DVM) running under a unique UNIX uid. The phone comes pre-installed with a selection of *system applications*, e.g., phone dialer, address book

1) Secure Access through Two-Factor Authentication:

Two-factor authentication has replaced the use of username and passwords as a secure non-repudiated model to control access. Two-factor authentication, often referred to as "strong authentication," replaces the use of a single "secret," the password, with the combination of two or more of the following factors:

All rights reserved by www.ijsrd.com

- Something you have (such as a token or a card)
- Something you know (such as a PIN)
- Something you are (such as a biometric fingerprint image)

Authentication Solution	Risks	Pros and Cons
Username and Passwords	Risk Level: HIGH If not properly managed or protected, Username and passwords can be easily stolen and provide easy access to your network or systems.	Pros: • Easy to implement and commonly used for both network and system access. • Users are more familiar with Username and password systems than any other authentication system. Cons: • Passwords can be guessed if based on common words or names. • Username and passwords can be easily stolen with freely available hacking tools, or by Trojans and key-stroke loggers.
Digital Certificates	Risk Level: MEDIUM Digital Certificates stored on a user's desktop can be stolen or spoofed.	Pros: • Behind the scenes system that is passive and invisible to the user. • Requires no action on the user's part. Cons: • The distribution and implementation of digital certificates can be costly and require the set up of an internal PKI system.
Biometrics	Risk Level: MEDIUM (if used as single factor authentication) Depending on the fingerprint scanner that is being used, the possibility of copying the user's fingerprint data. There's also the possibility of replaying the stored digital data representing the biometric reading.	Pros: • Nearly impossible to steal an iris scan, face pattern or fingerprint and difficult to fake. • Best used as a second factor in a two-factor system to augment a Username/password or card-based system. Cons: • Requires significant hardware cost to implement. • Potential to spoof fingerprint image, or minutia data (depending on what system is being used) • The technology still isn't foolproof and is subject to false readings and significant user training. • 10-16% of users have finger images that are unreliable due to age, work or other environmental factors.

Fig. 1: Various security features

B. Various security features:

Android security is multi-layered and more robust than traditional Linux. This talk covers in-depth analysis of various security features introduced till Jellybean 4.3.

- Memory Protection Options
- Application Sand Boxing
- Full File system encryption
- Storing the master keys and certificates
- Sean droid project
- Full File System Encryption
- Cryptography
- Secure inter process communication
- Application signing
- Application-defined and user-granted permissions

C. Android Security Overview:

The genesis of the personal computer era gave natal to a new form of illicit, the hacker. These criminals, now cooperate with each other and operating more efficiently than ever, aim to adventure any system containing interesting or valuable data; often with the goal of self-righteousness and financial gain at the top of their minds. For the past few eras their focus has been on exploiting Microsoft Windows, largely due to the popularity of the operating system combined with the vast amount of personal data stored on PC's. However today there is a new even more smart target for hackers to exploit, the Android smartphone.

D. Android Security Risks:

1) AWOL Androids:

The top concern about any mobile device is loss. In a Juniper survey, 58 percent of smartphone and tablet users be anxious not being able to recover lost content. Apple iPhone users can refurbish nearly everything from iTunes, but Androids are not managed via desktop sync. Data loss can be avoided in two ways. First, install an auto-backup app (e.g., Wave Secure, My Backup) to enable quick refurbishment of all that matters to you. Second, enroll your Android with one of the many available "find me" services to locate and recover lost devices.

2) Flimsy passwords:

If your Android falls into the wrong hands, more is needed to prevent thieves from stealing broadband service, ringing up SMS fees, reading your email, or abusing VPN connections. In Juniper's survey, 3 out of 4 users locked their smartphones. This is an excellent first line of defense, but users need to understand Android's limitations.

3) Naked data:

A major business risk posed by Android is lack of hardware data encryption. Fortunately, Android 3.0 ("Honeycomb") adds an API to let constructers offer encryption and IT enforce use. Unfortunately, existing Androids cannot yet perform hardware encryption. Until self-encrypting Androids appear, stored data can be protected in two ways. First, those remote lock apps and APIs can request remote wipe as well, resetting the device to factory defaults – but only when reachable, without dabbing SD card data. For more rigorous protection, enterprises should scramble sensitive data such as email and contacts using self-encrypted apps (e.g., Good for Enterprise, Exchange Touchdown)

E. Android malware:

According to traffic analysis by Adaptive Mobile, Android malware spike 400 percent last year. The total is still miniscule compared to other platforms, but more malware is likely to target Android's rapidly-expanding pool of potential victims. When Coverity assessed the Android kernel, it identified 359 code vulnerabilities, 88 of which posed "high risk" of exploitation. Because Android is an open development platform, hackers have ample opportunity to find and learn how to take advantage of these kinds of flaws.

IV. CURRENT SMARTPHONE PLATFORMS

In this chapter we present an overview of current popular smartphone models, namely Android, Blackberry, iOS, Symbian and Windows Phone 7. For each, the description is broken down in terms of key security abstractions, in order to contrast the different mechanisms employed by each smartphone.

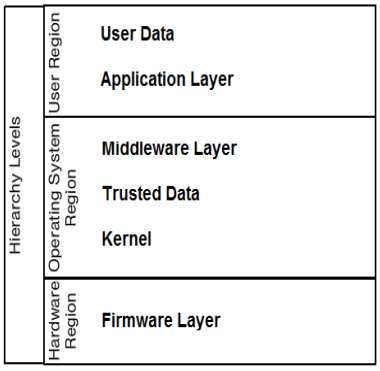

Fig. 3: Trusted computing environment

All rights reserved by *www.ijsrd.com*

A. Android application architecture:

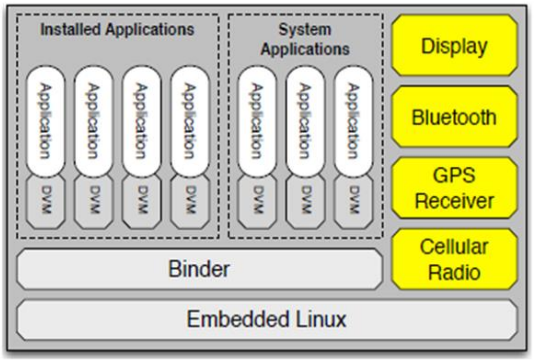

Fig. 4: Android application architecture

Analysis This section surveys a set of previously published exploits and vulnerabilities for Android and describes the results of analysis and testing performed to assess the effectiveness of SE Android in addressing the threats of flawed and malicious apps. It then provides a general discussion of the threats that can and cannot be mitigated by SE Android. The analysis and testing was performed using the initial SE Android policy configuration developed before reading about any of these specific exploits or vulnerabilities in Android. The policy configuration was developed based on normal Android operation and SELinux policy development practices.

B. Android Security Features:

In Android, on-kernel applications are executed in a virtual environment provided by the Dalvik Virtual Machine. Android uses the Dalvik Virtual Machine (DVM) with just-in-time (JIT) compilation to run Dalvik executable translated from Java byte-code. The DVM is similarto the Java Virtual Machine (JVM) in concept and as such, many traditional Linux and Java application testing techniques can be applied to Android.

C. Android version 4.0 (ICS) provides additional security features:

Such as full disk encryption and the support of Address Space Layout Randomization (ASLR); however, very few devices have this version of the OS installed by default at the time of writing.

D. Threat Agents:

It can berelevantto define the threat agents when assessing security measures and their associated risk. One of main threat is the individual attacker. Using obfuscation in a product can potentially be enough to protect against the casual attacker but an experienced attacker could haves specific skills dedicated to mobile exploitation such as the expertise to develop custom tools for application de obfuscation and decryption of data using a large GPU cluster.

Another threat is corporate espionage. Several companies made the news after reporting incidents of industrial espionage4. Corporate spies may not have direct contact with targeted devices, but they have access to large financial resources and highly skilled computer experts.

E. Touch screen:

Touch screens have been widely adopted recently as the solution for interacting with portable devices such as smart phones tablets, notebooks, navigation systems, and so on. Touch screens utilized in consumer portable devices are mainly add on Types where the touch screens are separated from the display panel. Design and manufacture of add-on type touch screens is a mature industry with many commercially available sensing methods. Common sensing methods include: resistive, capacitive, acoustic-wave, and

Steps	Description
1	You will use Eclipse IDE to create an Android application and name it as Login Screen under a package com.example.loginscreen. While creating this project, make sure you Target SDK and Compile With at the latest version of Android SDK to use higher levels of APIs.
3	Modify src/MainActivity.java file to add necessary code.
4	Modify the res/layout/activity main to add respective XML components
5	Modify the res/values/string.xml to add necessary string components
6	Run the application and choose a running android device and install the application on it and verify the results

Table 1: touch screen description

infrared based touch sensing techniques. Among these, the capacitive based method is increasingly popular because of its sensitivity, durability, and ability to detect multi-touches. The typical response time of a capacitive touch panel is 4ms.

F. Fingerprint Sensors:

The optical type of fingerprint sensor requires a lens system and is hard to implement in a small package at a low cost. One alternative is a TFT (thin film transistor) based fingerprint sensor. TFT technique is well known for creating large size displays. The technique puts ICs directly onto a glass substrate. It is the most cost effective and scalable way for creating fingerprint sensors that can cover larger area than the standard CMOS process based approach

V. DESIGN

A login application is the screen asking your credentials to login to some particular application. You might have seen it when logging into Facebook, twitter etc.

A. Actual design:

First you have to define two Text View asking username and password of the user. The password Text View must have input Type set to password. Its syntax is given below:

Security design:

```
<EditText
  android:id="@+id/editText2"
  android:layout_width="wrap_content"
```

All rights reserved by *www.ijsrd.com*

```
  android:layout_height="wrap_content"
  android:inputType="textPassword" />
<EditText
  android:id="@+id/editText1"
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
/>
```

B. Login screen code run in Emulator:

Fig. 5: Login Screen

In the java file, inside the method of on Click get the username and passwords text using get Text () and to String () method and match it with the text using equals () function

```
EditText username =
(EditText)findViewById(R.id.editText1);
EditText password =
(EditText)findViewById(R.id.editText2);
public void login(View view){
if(username.getText().toString().equals("admin") &&
password.getText().toString().equals("admin")){
//correcct password
}else{
//wrong password
}
```

C. User name password code:

Login screen:

Fig 6: Login Screen

VI. SYSTEM DESIGN

A. Why Android smartphones? :

The reason is simple; these devices contain more personal information than the average PC. When a smartphone app logs into your email account, it remembers the password so you never have to log in again. This behavior is common whether it is a personal email account, corporate VPN access, PayPal or Facebook.

B. Module Pictorial:

1) Representation:

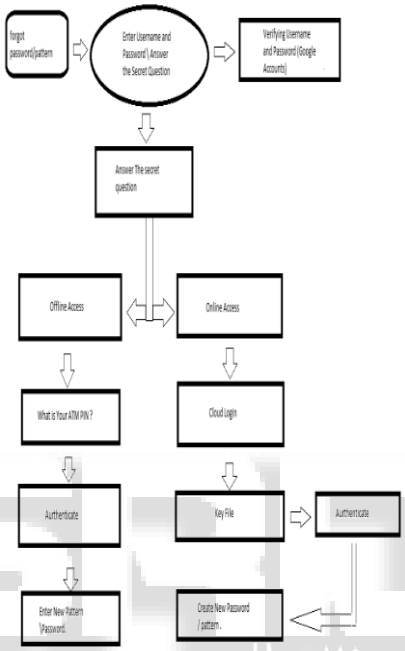

Fig. 7: Representation

C. PASSWORD MANAGEMENT:

1) SECURITY SENSITIVE SMARTPHONE APPLICATIONS:

The smartphone is already hosting a multitude of applications, and the security concern with each next generation application is growing. This is because every next application has improves on the smartphone's unique functionalities, but also increases the risk with information passing and storage.

2) Smartphone as a Server:

Using the smartphone as a server for personal usage is a natural direction for the smartphone, even more so than using it as a workstation. With the smartphone increasing in computational complexity, as well as housing

3) Application Level Malware:

For security mechanisms to work, it should be a viable assumption that the security services can rely on the kernel to supply correct data. Malware can entry to a device via social-engineer techniques or by communication vectors on the smart phone. Once there, a malware can remotely control a device, modify the system, Expose coned data, install rootkit, and breach the device security policy in general.

All rights reserved by *www.ijsrd.com*

After this, permission escalation is to maliciously using the permissions granted to an installed application. One attack that elected several smartphone was the Web kit web browser attack, which was done through a buffer overow in an outdated native library and a cross-site scripting vulnerability, which allowed the hacker to run malicious code on the device using the browsers high privileges. This was able to be done and lead to a user space jailbreak (complete control of the system) because

4) Kernel Level Malware – Rootkits:

Kernel level malware exploit a vulnerability in the operating system kernel or System libraries. Rootkits can do malicious activities such as stealthily placing aCall, listening into confidential conversation, read and send location data, and drain Resources such as battery [42]. At the microkernel level, the rootkit gains full access of the target system, by being able to inspect all communication between the operating system and the hardware, as well as evade detection with this more control

5) Insecure Data Transfer:

Malware infections can spread from a smartphone to other devices that it is connected to via peripheral links and vice versa, showing a crossing-over behavior. In 2005 Cardtrap.A was a Symbian SIS _le Trojan not only disabled application on the Cell phones, it also installed three Windows worm on the device's memory card which would move to the workstation once the card in inserted in it [36]. In 2006 Crossover virus moved from the Windows workstations to Windows Mobile Pocket PC [36].

VII. SMARTPHONE HACKS, ATTACKS AND JAIL BREAKING

All code has a nonzero probability of containing vulnerabilities and although minimizing threats and patching known vulnerabilities prevent security failures, it does not mitigate the amount of damage an attacker could indict once a vulnerability is found. In this chapter, we duplicated a couple of simple but severe attacks, which help us understand common security negligence which can severely compromise the security framework of the smartphone.

A. SMARTPHONE SECURITY POLICIES

On any smartphone device the parties each with some guarantee of access are the smartphone owner(s), the person presently in access of the smartphone, application framework developers, third party application developers, operating system developer/community, local subnet (if open to Bluetooth or Wi-Fi area), the device manufacturer, network carrier/service provider and the government which owns the network bandwidth.

1) Password Usage:

Let's be honest, passwords are annoying. These days, we need a password or PIN everywhere. We have so many that we can't keep track of them all. We forget to update them and when we do, it's difficult to come up with effective ones that we can still remember, so we procrastinate changing them for months, even years. We all know this is bad, but the alternative the painful irritating password creation and memorization process is sometimes more than we can tolerate. There is hope! Passwords don't have to be complex cryptograms. A few simple methods can help make living with passwords a little easier.

B. creating good passwords:

A good password is one that is easy to remember but difficult to guess. That sounds like a paradox, but it's really not.

There are a couple of different ways to create difficult-to-crack passwords. One is substituting letters with characters and numbers. To make it easier on yourself, try to use numbers and characters that resemble the letters they are replacing.

C. Know what makes for a bad password:

Because the attacks described above are becoming increasingly more common, you don't want to use anything in your password that's personal and easy to guess. Keep in mind the following don'ts:

- Don't use only letters or only numbers.
- Don't use names of spouses, children, girlfriends/boyfriends or pets.
- Don't use phone numbers, Social Security numbers or birthdates.
- Don't use the same word as your log-in, or any variation of it.
- Don't use any word that can be found in the dictionary even foreign words.
- Don't use passwords with double letters or numbers.

VIII. ANDROID HACKING TRICKS

A. SMS Trojans:

Android malware has evolved its tactics and distribution over the last two years. Two big news makers for Android malware were Trojans SMS, a premium-SMS Trojan, and DroidKungFu, a boot with rooting capabilities

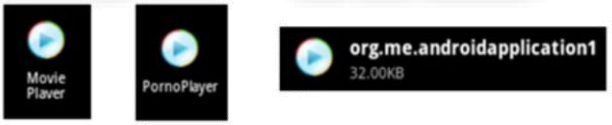

Fig. 8: SMS Trojan

```
{
  localSmsManager.sendTextMessage("3353", null, "798657", null, null);
}
catch (Exception localException2)
{
  try
  {
    localSmsManager.sendTextMessage("3354", null, "798657", null, null);
  }
```

Fig. 9: Trojan code

B. Rootkits:

We have encountered numerous examples of Trojans with root capabilities. These Trojans often have command & control functionality similar to what has been seen with PC botnets. Because these apps root, they gain escalated privileges and are able to bypass Android's permission model; thereby granting access to all functionality on the device without user notification. Taking advantage of known exploits in the Android OS, malware authors bundle these exploits in their APK's. The rooting exploits are the same ones made available by hackers for those willing to intentionally root their device. The two most prevalent ones

All rights reserved by *www.ijsrd.com*

target versions 2.1 and 2.2 of the Android OS, rage-againstthecage and exploit.

C. Spyware:

Other types of threats are those that spy on you or steal your data. There are a number of apps that are the equivalent to commercial key loggers found on PCs. These apps offer their services to 'track' your kids, spouse or employees. These behaviors are easy to incorporate into an app and this begins with the easy task of requesting the necessary permissions.

```
public void callrecord()
{
  try
  ......
    {
      MediaRecorder localMediaRecorder = new MediaRecorder();
      this.recorder = localMediaRecorder;
    }
  .......
public void onCreate()
{
  super.onCreate();
  PowerManager.WakeLock localWakeLock =
      ((PowerManager)getApplicationContext().getSystemService("power")).newWakeLock(1, "RecordService");
  .....
    String str = localSimpleDateFormat.format(localLong);
    this.filetime = str;
    stopCallRec();
    callrecord();
  }
}
```

Fig. 9: Android spyware

IX. ANDROID MALWARE DISCOVERY TECHNIQUES:

Identifying Android malware can be a challenging task for researchers. Many times, identifying an app as malicious based on high level data can be misleading and lead to complications in the research process if the researcher is not positive of what they are looking for. The permission model is a good place to start; but because an app requests a certain permission does not make it malicious. Often times, we must dig deeper to find out the true intent of the app. Digging deeper can be automated by utilizing static and dynamic tools.

A. safeguard your password:

At first, it may be difficult to remember your password. Did you substitute an "i" with a "1" or did you use a "1" to represent "L?" Most people will want to write the password on a piece of paper and place it underneath their keyboard or mouse pad. Or worse, they'll stick the password right on their monitor.

B. Password Cracking:

While passwords are a vital component of system security, they can be cracked or broken relatively easily. Password cracking is the process of figuring out or breaking passwords in order to gain unauthorized entrance to a system or account. It is much easier than most users would think. (The difference between cracking and hacking is that codes are cracked, machines are hacked.) Passwords can be cracked in a variety of different ways.

C. Access Control:

Access control is a system that control access to resources. Access control is required in smartphones to monitor activities so that congeniality, integrity and availability is maintained. There are three type of access control that we can use alone or together in security policies for smartphones. Mandatory Access Control (MAC) is rule based access control enforced by the operating system and unalterable by the subjects or owner of objects [13, p.103].

X. OFFLINE ACCESS:

A. How to Choose Good Passwords:

Now that we have established the importance of passwords and some of the ways in which they may be vulnerable to cracking, we can discuss ways of creating good, strong passwords. In creating strong, effective passwords it is often helpful to keep in mind some of the methods by which they may be cracked, so let's begin with what NOT to do when choosing passwords.

B. No Dictionary Words, Proper Nouns, or Foreign Words:

As has already been mentioned, password cracking tools are very effective at processing large quantities of letter and number combinations until a match for the password is found, as such users should avoid using conventional words as passwords. By the same token, they should also avoid regular words with numbers tacked onto the end and conventional words that are simply written backwards, such as 'nimda'. While these may prove to be difficult for people to figure out, they are no match for the brute force attacks of password cracking tools.

C. No Personal Information:

One of the frustrating things about passwords is that they need to be easy for users to remember. Naturally, this leads many users to incorporate personal information into their passwords. However, as is discussed it is alarmingly easy for hackers to obtain personal information about prospective targets. As such, it is strongly recommended that users not include such information in their passwords. This means that the password should not include anything remotely related to the user's name, nickname, or the name of a family member or pet. Also, the password should not contain any easily recognizable numbers like phone numbers or addresses or other information that someone could guess by picking up your mail.

D. Length, Width and Depth OF PASSWORDS:

A strong, effective password requires a necessary degree of complexity. Three factors can help users to develop this complexity: length, width & depth. Length means that the longer a password, the more difficult it is to crack. Simply put, longer is better. Probability dictates that the longer a password the more difficult it will be to crack. It is generally recommended that passwords be between six and nine characters. Greater length is acceptable, as long as the operating system allows for it and the user can remember the password. However, shorter passwords should be avoided.

XI. CHANGING & STORING PASSWORDS AND PINS

In order to ensure their ongoing effectiveness, passwords should be changed on a regular basis. Changing passwords securely is fairly simple. Windows passwords are changed through the Control Panel and in UNIX, the 'pass word' command generally does the trick. A good rule of thumb is to change passwords as close to the actual account as possible. For example, if it's an ISP account, don't telnet through three other machines to change that password. If it's

All rights reserved by *www.ijsrd.com*

an office computer, users should be on that computer and not on a co-worker's when changing it. Don't let anybody watch while typing the old and new passwords. If at all possible, the password should be changed over a secure connection like a secure shell (SSH).

A. Change your password often as in several times a year:

Your network administrator can force your employees to change their password every so often. Microsoft recommends having users change their passwords every 30 to 90 days, but encourages you to go with the smaller number. I think 30 days is a reasonable number here. You always want to side with caution when it comes to sensitive information.

XII. ONLINE ACCESS

A. Email-ID:

NFC is also useful in simplified transactions, data exchange, and connecting electronic devices with a touch. As promoted [29] NFC devices can read NFC tags on a museum or retail display; and can pair with Bluetooth and share contacts, photos, songs, applications or videos.

B. Remote Lock:

Already cars (such as in the case of the 2011 Chevrolet Cruze) have started featuring with a smartphone enabled remote lock that allows the owner to check the fuel gauge, lock and unlock the car, set as the horn and lights alarm and perform onboard diagnostics, such as checking tire pressure, by remote.

C. Network Administrators Password:

Managers and administrators can enhance the security of their networks by setting strong password policies. Password requirements should be built into organizational security policies. Network administrators should institute by regular changes/updates of passwords. They should also regularly remind users of how easy it is for hackers to get their passwords through social engineering and online attacks. New users should be taught about good password practices. Providing intranet resources on network security and password security can also be helpful. Finally, the organization's password policy should be integrated into the security policy, and all readers should be made to read the policy and sign-off on it.

Authentication Solution	Risks	Pros and Cons
One Time Password (OTP) tokens	Risk Level: LOW The possibility of a Man in the Middle attack that steals the Username, the PIN and the value from the OTP token.	Pros: • Easy to use system requiring only a small token displaying a changing PIN or password. • Provides an extra layer of security to a Username and password. Like a Username and password, can be used for both network and system access. Cons: • Implementation and device cost • Used primarily for remote access. Integration with various applications is costly and inconvenient for users. • Susceptible to man in the middle attacks. • If the Username and password are compromised and the token is stolen, a malicious user has full access to the system/network. • Without the device, the user cannot login to their system without bypassing the security model and utilizing help desk resources
Contactless Smart Cards	Risk Level: LOW The possibility of tampering with the card's chip to obtain login information.	Pros: • Most users are already carrying a contactless or proximity badge in the form of a corporate ID and are using this badge to access the physical facility. Cons: • Recently introduced into the market for logical access. • Different logical access security models based on what technology is used.
Contact Smart Cards	Risk Level: LOW If both the card and the PIN are stolen, undetected unauthorized access is possible.	Pros: • Smart Cards are portable and easy to integrate into a two-factor authentication system. They can be used for either network or system access. • They can safely hold and store lots of data, including encryption keys and other user authentication information. Cons: • Still not widely used due the effort and cost to install and manage the PKI to support smart cards.

Fig. 10: Network Administrators Password

XIII. IN CONCLUSION

All operating systems have distinct strengths and weaknesses; however, many are the same and essentially are up to the user and the configuration of the password. Users need to remember not to install apps from unnecessary sources, especially if they are unknown. While users can't know them all, users need to ensure that they are from a reputable source. If not, that is where malware commonly comes from, with backdoor apps masquerading as secure applications. Also, jail broken phones are at a huge risk if the user maintains the default password and an even higher risk if not used in the Apple marketplace. Instances of malware exist on all of the phones and are even more relevant on ones using untrusted app sources. Consumers can keep this research in mind when using their smartphone to best protect their valuable information.

REFERENCES

[1] Armstrong, Del and Simonson, John: "Password Guessing" and "Password Sniffing," An Intro to Computer Security, School of Engineering & Applied Sciences, University of Rochester, Oct. 25, 1996. http://www.seas.rochester.edu:8080/CNG/docs/Security/security.html

[2] Belgers, Walter: "UNIX Password Security," JANET-CERT, Dec. 6, 1993. http://www.ja.net/CERT/Belgers/UNIX-password-security.html

[3] Donovan, Craig: "Strong Passwords," SANS Institute, June 2, 2000.http://www.sans.org/infosecFAQ/policy/password.htm

[4] Wilner Nina (2009) " Android On Power Architecture", ELC, Grenobles

[5] B. M. Bowen, P. Prabhu, V. P. Kemerlis, S. Sidiroglou, A. D. Keromytis, and S. J. Stolfo. BotSwindler: tamper resistant injection of believable decoys in VM-based hosts for crimeware detection. In Proceedings of the 13th international conference on Recent Advances in Intrusion Detection, 2010

[6] M. Bishop, Computer Security. Addison Wesley, 2003.

[7] Shabtai, Y. Fledel, U. Kanonov, Y. Elovici, and S. Dolev, \Google Android: A State-of-the-Art Review of Security Mechanisms," Neural Networks, p. 42,2009.

[8] Shabtai, Y. Fledel, U. Kanonov, Y. Elovici, S. Dolev, and C. Glezer, \GoogleAndroid: A Comprehensive Security Assessment," Security Privacy, IEEE, vol. 8, pp. 35{44, March-April 2010.

[9] J.Anderson, J. Bonneau, and F. Stajano, \Inglorious Installers:Security in the Application Marketplace," 2010.

[10] Worldwide smartphone markets: 2011 to 2015 - analysis, data, insight And forecasts. http://www.researchandmarkets.com/research/7a1189

[11] worldwide smartpho. R. Adler and P. Desmares. An economical touch panel using saabsorption. *Ultrasonics, Ferroelectrics and Frequency Control, IEEETransactions on*, 34(2):195 –201, march 1987.

All rights reserved by *www.ijsrd.com*

[12] R.Aguilar and G. Meijer. Fast interface electronics for a resistive touchscreen. In *Sensors, 2002. Proceedings of IEEE*, volume 2, pages 1360 1363 vol.2, 2002
[13] Atmel. Touchscreen Controllers - Parameters. http://www.atmel.com/products/touchsolutions/touch screens/default.aspx, 2012.
[14] W.-S. Cheong, S.-M. Yoon, C.-S. Hwang, and H. Y. Chu. High-mobilitytransparent sno2 and zno-sno2 thin-film transistors with sio2/al2o3 gate Insulato rs. *Jpn J Appl Phys*, 48(4):04C090–04C090–4, apr 2009.
[15] N. L. Clarke and S. M. Furnell. Authenticating mobile phone users using keystroke analysis. *Int. J. Inf. Secur.*, 6:1–14, December 2006. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann. Touch me once and i know it's you!: implicit authentication based on touch screen patterns. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*.
[16] Facchetti and T. J. Marks. *Transparent electronics : from synthesis to applications*. Wiley, Chichester, U.K., 2010.
[17] T. Feng, Z. Liu, B. Carbunar, D. Boumber, and W. Shi. Continuous remote mobile identity management using biometric integrated touchdisplay. *45th Annual IEEE/ACM International Symposium on Microarchitecture Workshops (MICROW)*, 0:55–62, 2012.
[18] T. Feng, Z. Liu, K.-A. Kwon, W. Shi, B. Carbunar, Y. Jiang, and N. Nguyen. Continuous mobile authentication using touchscreen gestures. In *Homeland Security (HST), 2012 IEEE Conference on Technologies for*, pages 451–456, 2012.
[19] H. Hara, M. Sakurai, M. Miyasaka, S. W. B. Tam, S. Inoue, and T. Shimoda. Low temperature polycrystalline silicon 'tft' fingerprint sensor with integrated comparator circuit. In *Solid-State Circuits Conference - ESSCIRC 2004*, pages 403–406, 2004
[20] R. Hashido, A. Suzuki, A. Iwata, T. Okamoto, Y. Satoh, and M. Inoue. A capacitive fingerprint sensor chip using low-temperature poly-si tfts on a glass substrate and a novel and unique sensing method. *IEEE Journal of Solid-State Circuits*, 38, 2003.
[21] Scripting Layer for Android (SL4A), http://code.google.com/p/android-scripting/
[22] Mercurial Eclipse, http://javaforge.com/project/HGE
[23] J. Mantyjarvi, M. Lindholm, E. Vildjiounaite, S. marja Makela, and H. Ailisto. Identifying users of portable devices from gait pattern with accelerometers. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2005.
[24] P. Marcus, M. Kessel, and C. Linnhoff-Popien. Securing mobiledevicebased machine interactions with user location histories. In *Security and Privacy in Mobile Information and Communication Systems*. Springer Berlin Heidelberg, 2012.
[25] J.-Y. Ruan, P.-P. Chao, and W.-P. Chen. A multi-touch interface circuit for a large-sized capacitive touch panel. In *Sensors, 2010 IEEE*, pages 309 –314, nov. 2010.
[26] W. Shi, J. Yang, Y. Jiang, F. Yang, and Y. Xiong. Senguard: Passive user identification on smartphones using multiple sensors. In *IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications*, pages 141–148, 2011.

All rights reserved by *www.ijsrd.com*