

Introduction to Information Security
Prof. V. Kamakoti
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Lecture - 11

(Refer Slide Time: 00:10)

Simple Network Management Protocol

- SNMP is a mechanism for remotely administering and monitoring network devices such as switches, routers, etc.
- Read (and/or write) access to SNMP managed devices is controlled via SNMP "community strings" (e.g., passwords).
- **Because SNMP was developed long ago, and is often used to interact with relatively "simple" network devices, encryption was NOT a mandatory part of the protocol.** Thus SNMP (pre-SNMPv3) community strings are totally vulnerable to sniffing attacks, and unfortunately, deployment of SNMPv3 remains relatively limited.
- **Recommendation:** If you use SNMP, try to make sure it is SNMPv3, use community strings which aren't "public" and "private," use a dedicated out-of-band network for SNMP management and monitoring, and block off-site SNMP traffic (port 161 and 162, tcp and udp) at your border.

So, the next thing is about password again. The SNMP, the simple network management protocol; which monitors, remotely administers and monitors the network devices such as switches and routers. This SNMP was in place long years before. And, they use something that passwords. They use something called community strings which are used to read or write access the SNMP managed devices. Because SNMP was developed long long ago, and, is often used to interact with relatively simple network devices, encryption of a password was not a mandatory part of that protocol. The SNMPv3 has now made it a mandatory. So, the pre-SNMPv3 protocols, a community strings are totally vulnerable to sniffing attacks. And, so one of the recommendations here is that whenever you use SNMP, you make it sure that you have SNMPv3 and use community strings, which are private and use a dedicated out of band network for SNMP management and monitoring, and also block off-site SNMP traffic.

So, many of these words that I have uttered here will not make immediate sense to people who do not have understanding of networking. But, towards module four and five we will try and make these things more understandable to you. But at this point, I would

like to emphasize that SNMP and SNMP based devices, if you use a version of SNMP which is less than three, then you are in for a problem in terms of password vulnerabilities.

(Refer Slide Time: 02:31)

“Good News! We’ve Encrypted Our Wireless Links To Prevent Password Sniffing”

- Just for the record and as a matter of due diligence, **we all know that WEP really doesn’t offer any security against eavesdropping, and that WPA (TKIP) isn’t much better, right?** If you’re doing wireless encryption, these days you really need to be doing WPA (AES) or WPA2. (See, e.g., <http://www.smallnetbuilder.com/content/view/24251/100/> and <http://www.networkworld.com/news/2009/082709-new-attack-cracks-common-wi-fi.html>)
- Better yet, why encrypt traffic ONLY on your wireless network? Traffic on ALL network segments is vulnerable to being sniffed, so ALL traffic should be encrypted end-to-end ALL the time! Encrypting JUST wireless links leaves all your other traffic vulnerable to being sniffed!

108

The other thing is that people today think that last mail encryption is enough. Suppose, I am using a wireless LAN, so when I am on the system, that is, in your laptop or your desktop where you are going to enable the wireless, you have an encryption protocol. What does that ensure? Between you and the wireless access point, the encryption happens. Beyond that wireless access point, where there is a wire traffic, will it continue to have this encryption? It is not ensured.

But, there is a feel from the lay man who uses this. That when he puts a wireless encryption on his network card, wireless network card and it is sufficient the entire traffic is, the entire data that he is sending is encrypted which is not the case. When we enable an encryption on a wireless network card. It ensures encryption on the transmission of the data from that network card to the access point which is associated with it. And, nothing beyond.

Even for the encryption between the wireless access card and the access point, the wireless network card and the access point, there are different encryption mechanisms that are listed and used. For example, WEP, WPA (TKIP), etcetera. But, none of these are really that much secure. Today the best one is the WPA (AES) or WPA2. We want; so, these are the encryptions that you set on your wireless access card when you are sending

data to an access point. And so, for more details we have given a link to a website which can give you more details about attack and cracks on common Wi-Fi networks.

The more important thing is it is not for you to realize that when I encrypt the data which is sent wireless through the Wi-Fi, the encryption happens only till the access point. And beyond the access point, how the data is transmitted is not still known. And, so if I transmit the data through Wi-Fi, though I have put the most sophisticated encryption algorithm that encryption is between only, the card and the nearby access point. So, beyond that access point how the data is transmitted? If it is going to be unencrypted, still then the vulnerability that we have stated still exists.

(Refer Slide Time: 05:45)

“Evil Twin” Wireless Nodes

- Not to make you paranoid, but let me also bring up another wireless issue: how do you know that the wireless access point you' re connecting to is the *real* wireless access point you intended to connect to?
- You should be aware that a bad guy could potentially put up an “evil twin” wireless access point -- using the same SSID your production access points normally use -- and in most cases your users wouldn' t be able to tell the difference. If the bad guy can con you into using his node, he may be able to sniff all your traffic, including your passwords, even if your real nodes have been secured.
- 802.1x can help to address some of these issues, but deploying 802.1x is not painless

Another very interesting problem of the security is this “Evil Twin” problem. So, when we get associated to a wireless LAN, wireless network an access point is it that? So, let us say there is an access point called a one and therefore, I see on my screen when I want to associate, I see a one. Is it the same a one or it is some other a one? Which is masquerading or which is mimicking this a one. So, that is very important. So, there is a way by which it is shown that another access point can masquerade this a one. And, so instead of associating to the genuine a one, you go and associate yourself at the evil twin of this a one. And, why it is evil because once you get associated, rather than being the functionality of the normal access point which is basically a portal service, where it converts your wireless traffic to wire traffic and vice versa. This evil twin access point can basically start sniffing your data and do much more havoc. The 802.1x actually helps to address some of these issues, but the deployment of 802.1x is not painless.

(Refer Slide Time: 07:16)

4. Passwords Won't Get Changed

- Most sites encourage (read: “require”) users to change their password at least once every <N> months, where <N> might be as little as one month, or as much as twelve months or even more.
- There is often confusion about the origin and purpose of this periodic password change requirement, sometimes even among security staff, and some frankly view it as a pointless or even counter-productive policy (although folks still go through the motions to at least keep the auditors happy).
- I believe that periodic password changes are useful, and **SHOULD** be required.

Let me tell you some reasons why.

Another important reason why passwords are insecure is that though most of the sites encourage the users is to change the password. They say within N months, you should change it some k times. But, many people ask the question why we should change this password, what will happen if I keep it like this. And, they do not see or find the justification of why the password need to be changed. But, I believe that periodic password changes are actually useful and should be required. Let us go into some reasons of why should we change a password periodically.

(Refer Slide Time: 08:42)

Periodic Password Changes Limit The Window for Brute Force Attacks

- If you never change your password, an attacker conducting a brute force attack on your password may have a protracted window during which he or she can concentrate on cracking your password, confident that you haven't changed your password during that time.
- If you do change your password, the attacker will need to restart their cracking effort because cracking your old password typically won't help the attacker deduce your new password (unless your current password is derived from your previous password in some ascertainable way).
- The permissible time between changes is thus determined at least in part **by the length and complexity of the passwords** being used.

The periodic password changing will limit what you call as brute force attack. If the password is same, one can keep on trying my account for say some ten months and he

can find out that password. More the time I give for a single password, more easily somebody can do a brute force attack and get the password. If I keep changing it every day, then somebody cannot go.

So, the amount of time that brute force attackers gets is only one day. In one day for him to guess my password is going to be difficult. And, the next day I am going to change anyway. So, more the time I keep the password the same, the possibility that a simple brute force attack will identify my password is larger. So, that is why we need to change the password at regular intervals. What is that interval? That actually depends upon the length and the complexity of the password. If I have a large password or very complex password, then I may not change it for say one year or two year. If I have very simple password, then I may need to change it very quickly.

(Refer Slide Time: 09:47)

Password Changes Can Terminate Unknown Access by Parasitic "Silent Riders"

- Assume you have a user who buys a new computer, and sells their old one, **failing to nuke-and-pave its disk** before doing so.
- Also assume that the user has configured various applications to take advantage of saved passwords, perhaps for dialup, wireless, or VPN access.
- The purchaser of the old system thus gets not just a system, but also the former owner's network access, and will be able to continue using those saved credentials unless/until the former owner changes their passwords, thereby rendering the saved credentials invalid.
- If password changes don't take place, the parasitic silent rider can continue their unauthorized access forever...

Another important thing about password is that suppose I have a system and am logging into the network and I am actually asking the browser to save all these login information, password information. And, one day this system does not boot and then somebody comes and tries to retrieve, nothing happens. So, I try to discard that system. But, please note that I am going to discard along with the disk and I do not do anything with the disk. So, someone goes and takes the disk and retrieves the information, obviously he is going to get some information about the passwords that I have stored there; because I have used the web browser, etcetera to save all my password. So, the other person who is actually a silent rider can basically get access to the password. And, I suppose I do not change the password at all, the person who got access to the old system can coolly use that password

and access all my accounts. So, this also ensures that; this also is a very valid point that we should keep changing our password at regular intervals.

(Refer Slide Time: 11:08)

Password Changes Make It Harder For Users To Reuse Passwords on Multiple Sites

- Assume that at least some users will be prone toward using the same password on all the sites they frequent.
- Requiring periodic password changes may result in at least modest password diversity (following the change, the user may have the same password on all their sites except for the one that insisted on a password change).
- Of course, there's also the possibility that users will simply update ALL their passwords to the new value that one site may insist they pick, but sometimes you just can't keep people from shooting themselves in the foot. :-;

So, password changes make it harder for users to reuse passwords on multiple sites. So, we try and use only one password for all the sites because this is very difficult for us to remember a mapping between sites and passwords; which site uses which password. It is very difficult for someone to map this. So, requiring periodic password changes means that we need to keep changing passwords at different sites. So, all many people do; they have one password for all the sites they have access to. And, if one of the sites say that the password needs to be changed, they go and change the password for all the sites.

So, if a hacker gets this password, then he can go and hack all the sites with this password. So, that is the drawback of having a single password for all the sites one use. But, many times we cannot keep the people from shooting themselves in the foot.

(Refer Slide Time: 12:24)

Password Changes Make It Harder For Users To Hardcode-and-Forget Their Passwords

- If I don't think that I'll ever have to change my password, I may very well just save it in my browser and all my other applications, and then happily forget about it.
- On the other hand, if I know that every three months (or whatever), I'll have to change my password, and I know that I'll need to know my old password to pick a new one, I'm less likely to hardcode-and-forget my passwords.

Another thing is that many times people hardcode the password and forget the password. Hardcode in sense, they will ask the browser to remember and they will say, "Ok, I do not need to; the browser will remember". But on the other hand, if the user is asked to change the password every three months, and when changing the password I need to remember my old password. So, this hardcode and forgetting of password will reduce.

So, normally we do not tend to ask the web browser to remember the password; because if we go and ask the web browser to remember the password, then the old password would be forgotten by you. Since you have to change the password for every three months, you need to remember the old password. At least, the last old; last one password for your system and that will force you not to use this hardcode and forget the password. And, if you do not hardcode and forget the password, then the chances that it leaks because of a local storage reduces considerably.

(Refer Slide Time: 13:38)

Users Need to Know HOW They Should Legitimately Be Changing Their Password

- With phishing emerging as a growing problem at many sites, users may be continually presented with bogus requests to “confirm” their current password.
- One step toward hardening users to resist that sort of phishing is insuring that users know how they should be legitimately changing their password.
- Periodic password changes cause users to practice that process so they have an established mental baseline against which to compare illegitimate phishing efforts.
- Routinizing password changes also effectively prevents “ask the admin to change it for me” password changing channels simply because there’d be too many requests (the admins would quickly get tired of doing them!)

There is also the notion of some phishing attack which goes and says, you know, please change your password. They are very interesting attack. So, there will be a mail saying change your password and then he will start sniffing the network.

So, you receive a mail from a hacker, say change your password. And, from that moment he start sniffing the traffic that goes on your network. And, whenever you change the password, he knows what is the new password or hash of the new password and he can basically go and start cracking your password. So, one of the thing is every user should be educated on what is the duration by which they need to legitimately keep changing the password. So if a sudden request comes, which is before this duration, then this user can know that it is a spam message and not a genuine message.

(Refer Slide Time: 14:45)

Requiring Password Changes Provide A “Teachable Security Moment”

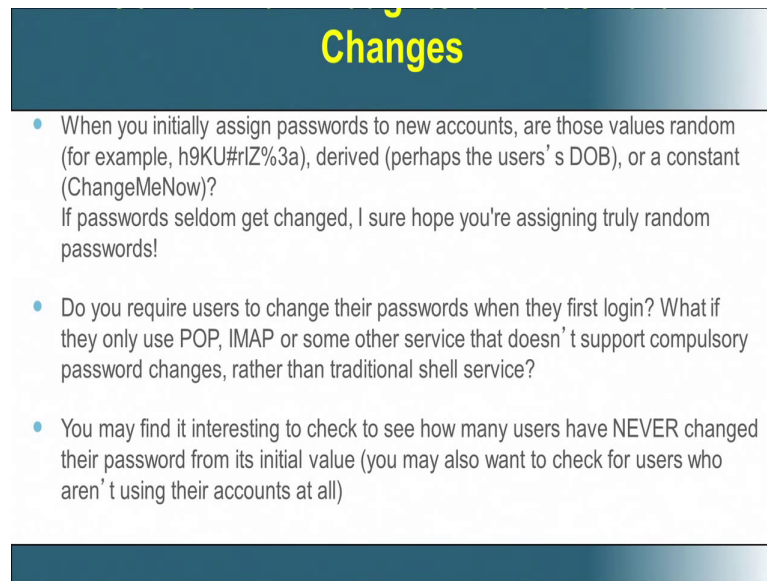
- Users need ongoing encouragement and education when it comes to system and network security.
- For example, users need reinforcement when it comes to basic security training, including things as basic as:

“Never disclose your password to anyone.”
- Most users, however, will not attend formal training sessions, and may never interact with security staff one-on-one. Things like quarterly password changes at least force users to think (at least a little!) about security at least a few times a year.

So, this requiring password changes actually provide what we call as a teachable security moment. People should understand that they are part of a secured society. And if something wrong they do that will cause lot of damage not only to them, but all the people, all the systems that are connected to them. So, many users do not come for training sessions related to security in many organization. So, things like you should change your password, you should not write your password, all those things, become unknown to many people who do not give a big care for sitting through training programs in some security or organization, who does not really care for security.

Now, if you say that every three months you need to change, the users now at-least remember once in three months that there is something called password and that password should not be disclosed and this is coming because of a security requirement. So, the word security will be registered at-least once in three months, whenever a user is forced to change his or her password. So, that is why we call this password change as a teachable security moment.

(Refer Slide Time: 16:13)



Changes

- When you initially assign passwords to new accounts, are those values random (for example, h9KU#rIz%3a), derived (perhaps the users' s DOB), or a constant (ChangeMeNow)?
If passwords seldom get changed, I sure hope you're assigning truly random passwords!
- Do you require users to change their passwords when they first login? What if they only use POP, IMAP or some other service that doesn' t support compulsory password changes, rather than traditional shell service?
- You may find it interesting to check to see how many users have NEVER changed their password from its initial value (you may also want to check for users who aren' t using their accounts at all)

So, the other thing is that whenever you initially assign passwords, they are some random values or some derived values like data of birth, etcetera. So, it is. So, when you open an account, first a default password is set and that password if it is genuinely random, then it is good. But, many times it is not random. It is derived from the user's data like the user's date of birth or some constant string like ChangeMeNow etcetera. We have seen about default passwords for many devices. But, this is default password when we try and open an account in some remote server. There are many services like POP, IMAP, where you never even you go and login again. So, many times this password remains the same with that default value. And if remains the same, then it is actually a threat for password stealing.

(Refer Slide Time: 17:35)

Email As a Password Reset Channel

- We're all familiar with sites that allow you to send a one-time password reset link to another email account (which you provided at the time you signed up).
- Does that process feel secure to you? It shouldn't...
- There are many ways that a bad guy could exploit that sort of password reset mechanism to steal your credentials, including:
 - sniffing (unencrypted) email to get a reset link (miss the first one? just ask for it another time...)
 - hijacking email traffic for a site by injecting a bogus DNS MX record (see Dan Kaminsky's "forgot my password" scenario/discussion at slides 24-39 of <http://www.blackhat.com/presentations/bh-dc-09/Kaminsky/BlackHat-DC-09-Kaminsky-DNS-Critical-Infrastructure.pdf>)

So, this is very similar to what we call as a one-time reset link. Right. So, whenever you ask for a reset and you go and click that reset, so an unencrypted email is sniffed, which is; see, see the request for a reset. You go and ask a server to reset your password and for that it will send you a link for which you can use to go and reset the password. Now, this email comes to you in an unencrypted form. So, a hacker who is watching your traffic will know that you are now trying to reset a password for a particular link. From that point of time, if he starts monitoring a network he can actually go and find out. If he starts hacking into your network; if he starts sniffing your network, he can find out what password you are going to set. So, there is a very very interesting discussion in the url that we have given in the pdf file, whose url we have shared here which talks about forget my password and what happens next.

(Refer Slide Time: 19:08)

“Trivial Pursuit” Passwords Resets

- Other sites allow password resets if the user can successfully regurgitate what I call “personal trivia.”
- Classic examples of “personal trivia” include:
 - date and/or place of birth
 - mother’s maiden name
 - last four digits of your social security number
 - your driver’s license number
 - name of your first employer
 - name of your favorite dog or cat
 - favorite flavor ice cream, etc., etc.
- Unfortunately, a lot of personal trivia answers are
 - (a) easily forgotten; or
 - (b) are a matter of public record due to things like genealogical databases, and social networking sites; or
 - (c) have low information entropy (e.g.: favorite ice cream flavor? Vanilla’s a safe bet)

Password reset is also not a very difficult task many times because the questions asked for resetting a password is also very trivial. And, it could be guessed. For example, the date or place of birth, mother's name, father's name, social security number, last four digits of your social security number, etcetera, your first employee or your favorite dog or cat, your favorite ice cream. Many of these informations are there in many of the social networking sites like your face book. I had a wonderful vanilla ice cream today, then they know that your favorite flavor is vanilla. And, these are very low information entropy because majority if you say again ice cream flavor, mostly the answer can be vanilla, sometimes strawberry and in some rare cases say butterscotch.

There are also some things which are geneological databases; which has the complete history of your entire family tree. From this, you can go and easily guess your birth date, your mother's name, your mother's second uncle’s, third daughter's husband’s name, anything could be guessed from this geneologicaltree. So, the way password gets reset, allows; there are two ways. One is a reset link and then somebody can sniff this email transactions by which you go and reset a password or it can also be through handling some trivial questions, which again anybody can guess answer and reset your password and get to know your password.

(Refer Slide Time: 21:18)

6. The Ten-Ton Gorilla in the Room

- Even if passwords had no technical security issues or practical security issues, there are policy driven reasons why passwords are no longer good enough.
- For example, consider the Payment Card Industry Data Security Standard (PCI-DSS) 8.3 (Version 1.2.1, July 2009). It requires:

"Incorporate two-factor authentication for remote access (network-level access originating from outside the network) to the network by employees, administrators, and third parties."
- Two factor authentication is in your future.

So, passwords; so to conclude here, passwords do have problems. And, the way these problems could be addressed is to have what we call as a two factor authentication. One of the important data security standard which is the payment card industry, data security standard, essentially says that one level authentication is not enough and they insist on a two factor authentication.

Thank you.