Windows Passwords: Everything You Need To Know

Jesper M. Johansson
Enterprise Security Architect
Security Business and Technology Unit
Microsoft Corporation
jesperjo@microsoft.com

Overview

- How passwords are stored
- How passwords are used
- How passwords are attacked
- Password best practices

How Windows Stores Passwords

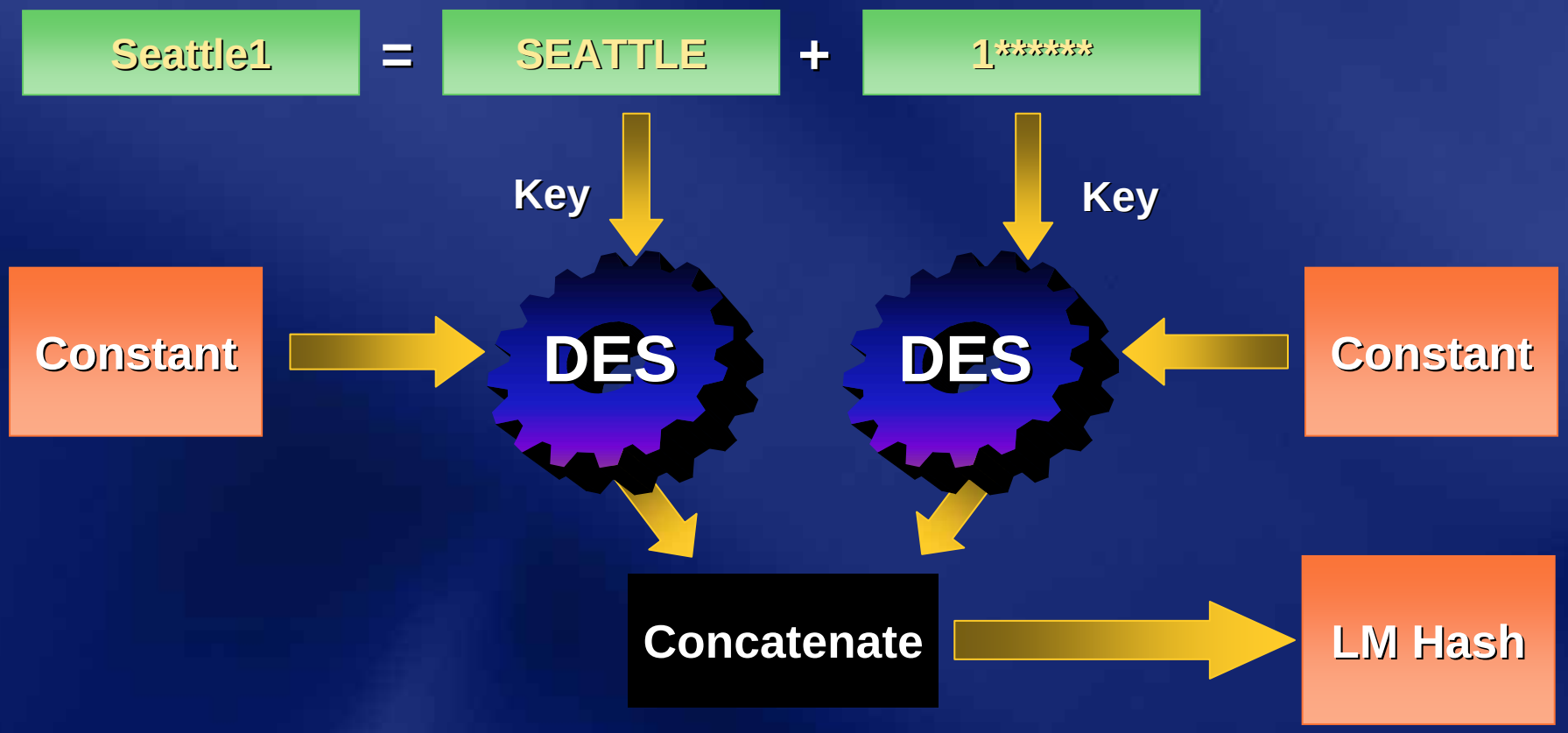
In the beginning…

Password Representations

- LM “hashes”
  - Old technology used on LAN Manager
- NT hashes
  - A.k.a., Unicode password or MD4 hash
  - Used for authentication on more recent Windows systems
- Cached credentials
  - Derivation of NT hash
- Stored User Names and Passwords
  - Calling application decides on representation

LM "Hash" Generation

- Padded with NULL to 14 characters
- Converted to upper case
- Separated into two 7 character strings



LM “Hash” Considerations

- It’s not a hash
- Limited character set
  - Common alphanumeric set only
  - Case insensitive
  - 142 symbols
- Padded to exactly 14 characters
  - Actually two seven-character passwords
- Maximum number of passwords ≈ $6.8*10^{12}$
- Unsalted…

Salting

- Prevents deriving passwords from password file
- Stored representation differs
- Side effect: defeats pre-computed hash attacks

```
Alice:root:b4ef21:3ba4303ce24a83fe0317608de02bf38d
Bob:root:a9c4fa:3282abd0308323ef0349dc7232c349ac
Cecil:root:209be1:a483b303c23af34761de02be038fde08
```

Same Password

NT Hash Generation

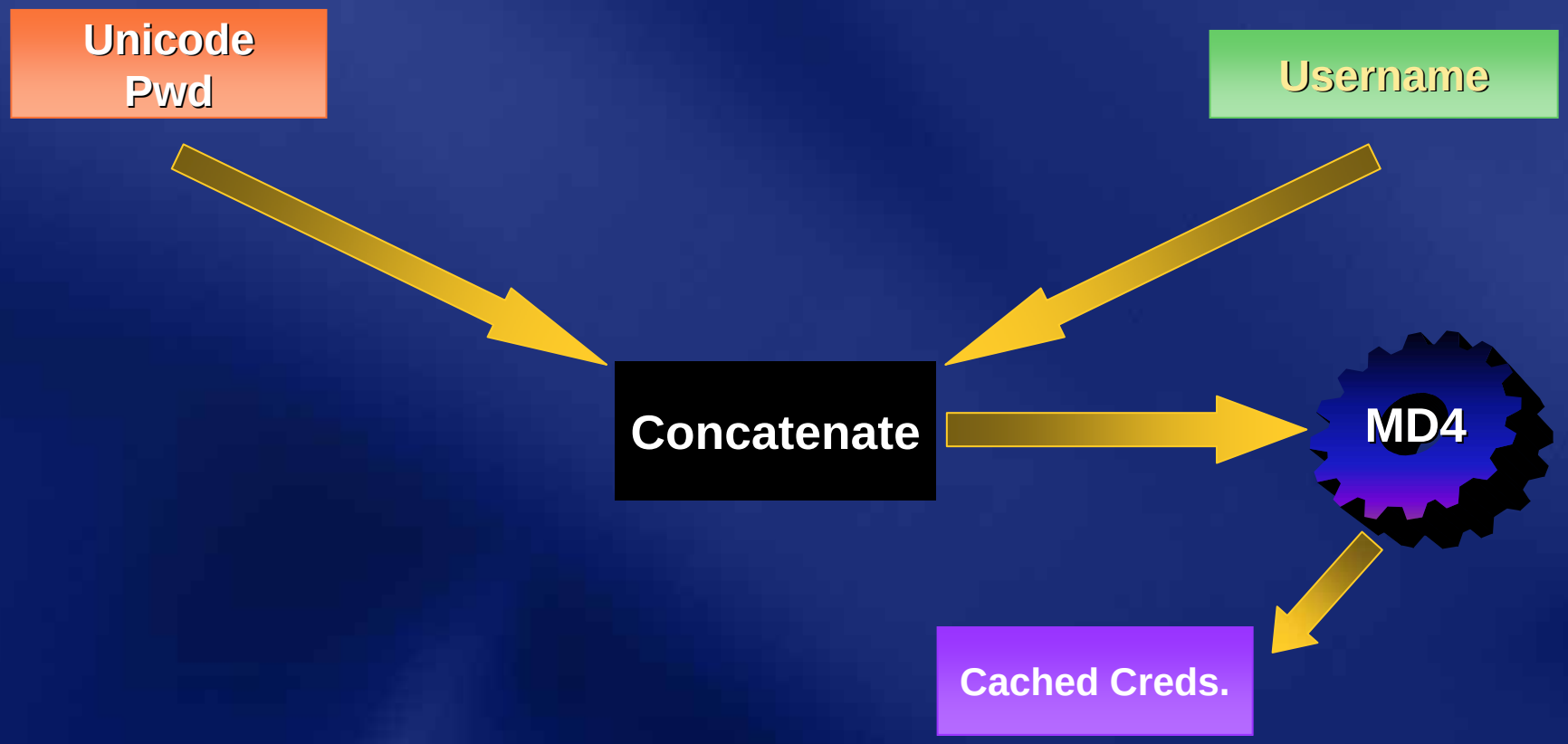
- Hash the password
- Store it

NT Hash Considerations

- Case preserving
  - 65,535 symbols
- Maximum length = 127 characters
- Number of ≤14-character passwords, same char set as LM hash ≈ $4.6*10^{25}$
- Number of ≤14-character password (full char set) ≈ $2.7*10^{67}$
- Number of 127-character passwords ≈ $4.9*10^{611}$
- Unsalted

Cached Credentials Generation

- Stored at logon
- Managed by LSA
- Hash of a hash



Stored User Names And Passwords

- Credential Manager
- Stores specific password-based credentials locally
- Applications can leverage for password storage
- Uses DPAPI for storage

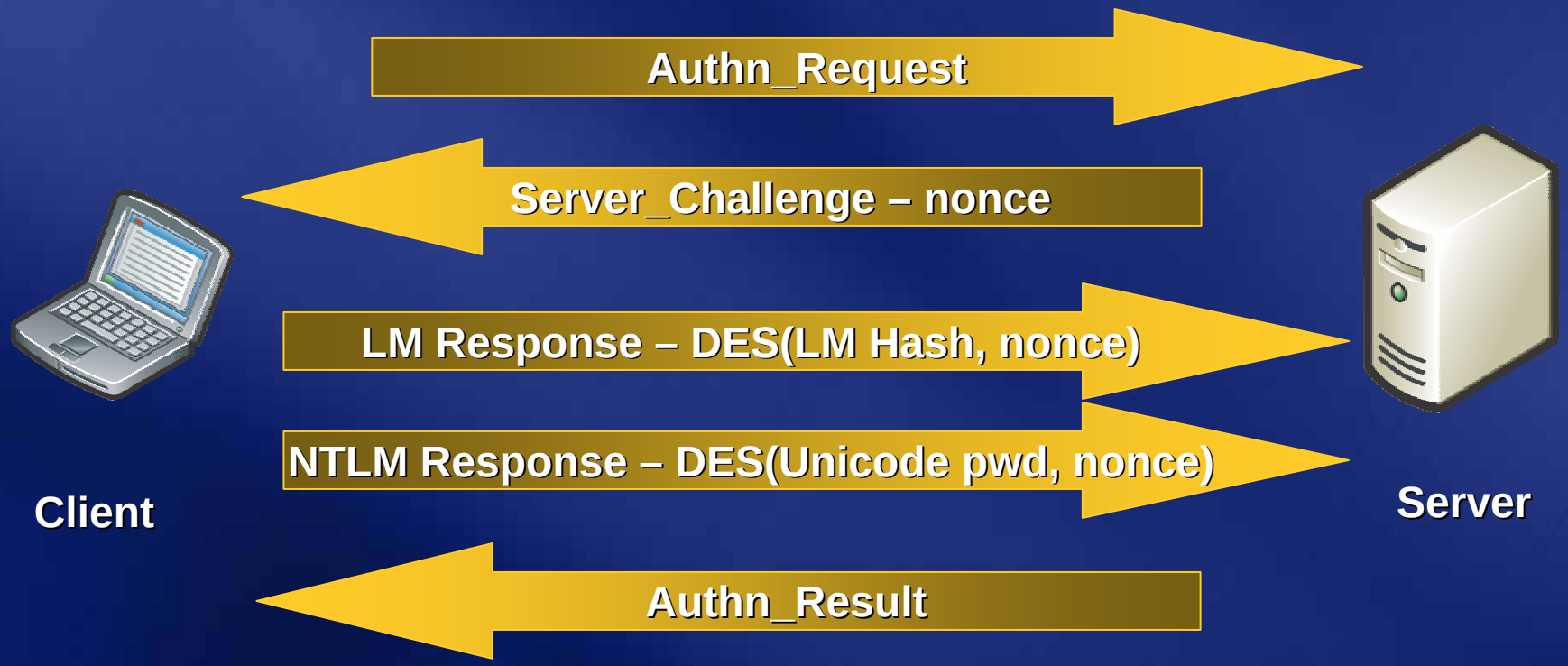
How Passwords Are Used

Authentication

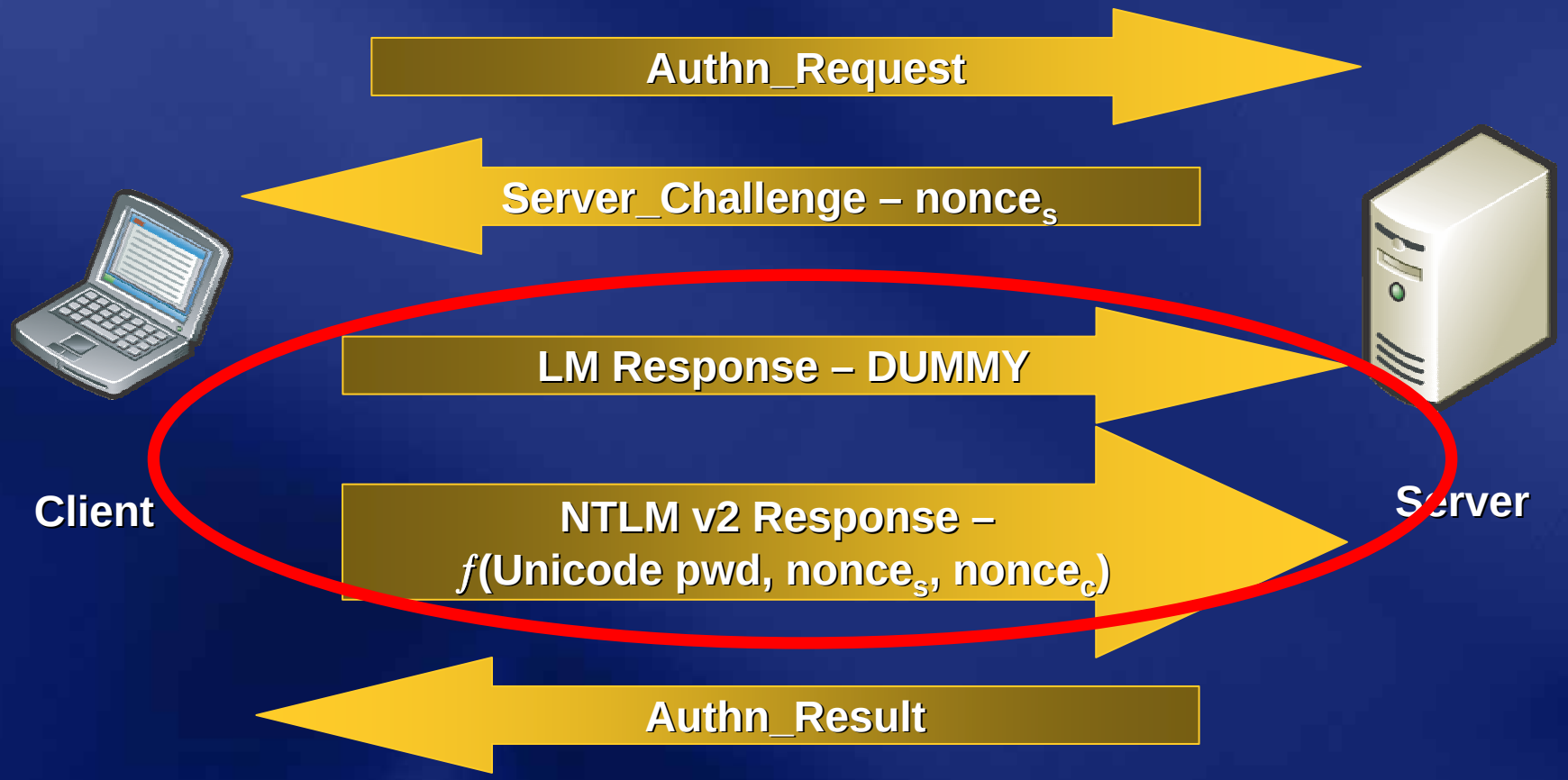
Authentication (authn)

- Winlogon passes the authn information to LSASS
- LSASS determines the authn package
- Local or remote login? If remote
  - Kerberos
  - MSV1_0
    - NTLMv2, NTLM, LM
- The chosen package generates authn data

NTLM And LM Authentication On The Wire



NTLMv2 Authentication On The Wire



LMCompatibilityLevel

Client-side impact

Level	Sends	Accepts	Prohibits Sending
0*	LM, NTLM,	LM, NTLM, NTLMv2	NTLMv2, Session security
1	LM, NTLM, Session security	LM, NTLM, NTLMv2	NTLMv2
2*	NTLM, Session security	LM, NTLM, NTLMv2	LM and NTLMv2
3	NTLMv2, Session security	LM, NTLM, NTLMv2	LM and NTLM

Server-side impact

Level	Sends	Accepts	Prohibits Accepting
4	NTLMv2, Session security	NTLM, NTLMv2	LM
5	NTLMv2, Session security	NTLMv2	LM and NTLM

* Default on some OS

Kerberos Authentication

- Authenticates access to domain resources by domain members
- Uses different operations than NTLM
  - Sensitive data is better protected from eavesdropping
- RFC compliant (yes, it is!)
- Uses NT hash
- Well documented

How Passwords Are Attacked

Key Point

- Bad passwords get broken, even when using good storage and authentication methods!
- Solutions
  1. Use better passwords
  2. Don’t let bad guys get the hashes

Four Types of Attack

- Passive online
- Active online
- Offline Attacks
- Non-electronic attacks

Passive Online Attacks
Wire Sniffing

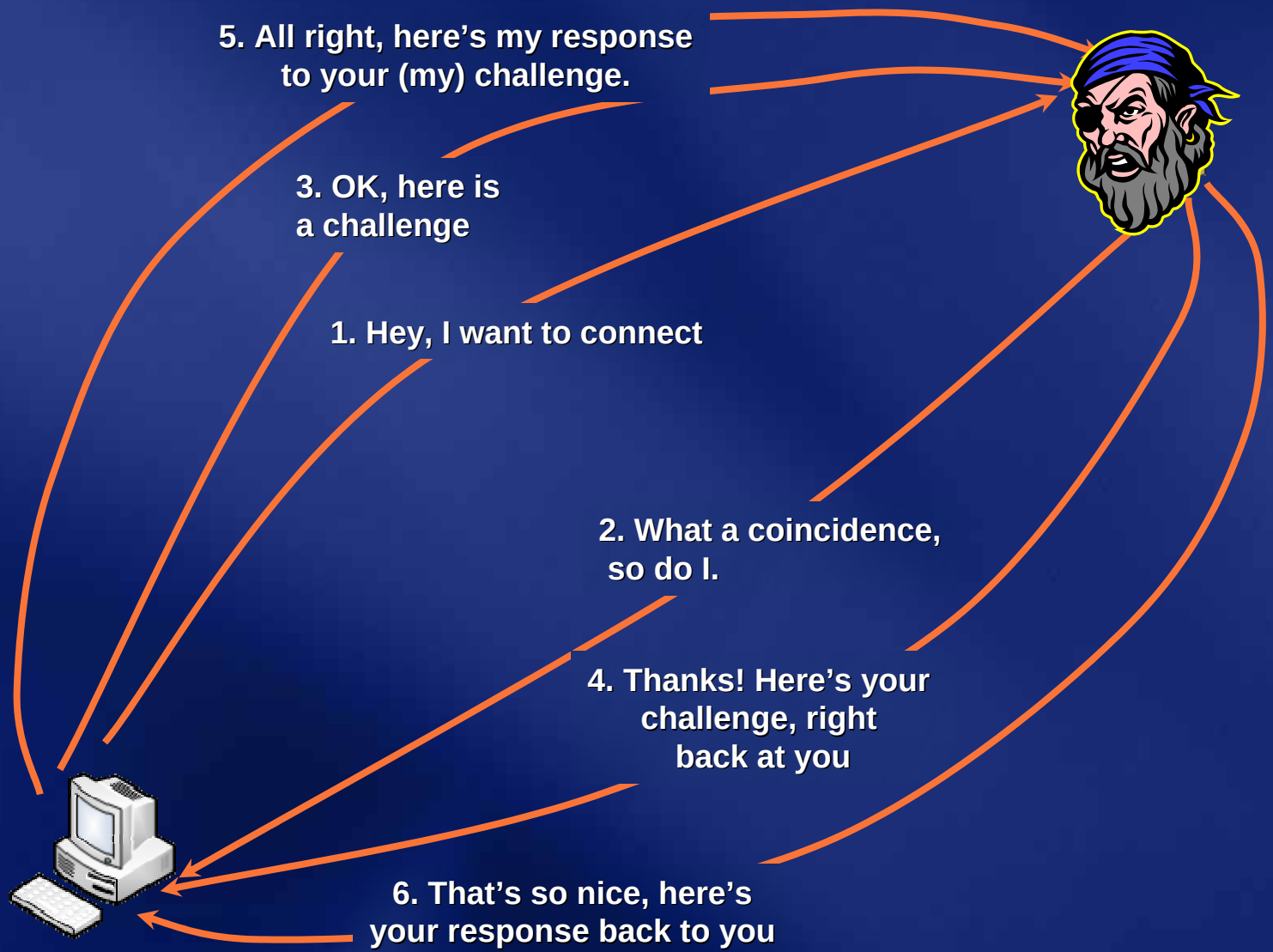
- Access and record raw network traffic
- Wait until authn sequence
- Brute force credentials
- Considerations
  - Relatively hard to perpetrate
  - Usually extremely computationally complex
  - Tools widely available

Passive Online Attacks

Man-in-the-Middle and Replay Attacks

- Somehow get access to communications channel
- Wait until authn sequence
- Proxy authn-traffic
- No need to brute-force
- Considerations
  - Relatively hard to perpetrate
  - Must be trusted by one or both sides
  - Some tools widely available
  - Can sometimes be broken by invalidating traffic

SMB Reflection Attack



Cracking v. Guessing

- Guessing from the logon prompt
  - Very slow
  - Easy to detect
  - Core problem: bad passwords
- Cracking presumes attacker has hashes
  - Hashes may be world readable
  - If not, system has already been hacked
  - Very fast
  - Core problem: bad guys with access to hashes

Active Online Attacks

Password guessing

- Try different passwords until one works
- Succeeds with…
  - Bad passwords
  - Open authentication points
- Considerations
  - Should take a long time
  - Requires huge amounts of network bandwidth
  - Easily detected
  - Core problem: Bad passwords

Offline Attacks

- Attacker has password database
  - How? Hard on Windows, easier on Unix
- Can attack at leisure
- Password representations must be cryptographically secure
- Considerations
  - Moore's law
  - Attacks against cached credentials about 3x slower

Offline Attacks

Dictionary Attack

- Try different passwords from a list
- Succeeds only with poor passwords
- Considerations
  - Very fast
  - Core problem: Bad passwords

Offline Attacks

Hybrid Attack

- Start with Dictionary
- Insert entropy
  - Append a symbol
  - Append a number
  - …
- Considerations
  - Relatively fast
  - Succeeds when entropy is poorly used

Offline Attacks

Brute-force Attack

- Try all possible passwords
  - More commonly, a subset thereof
- Usually implemented with progressive complexity
- Typically, LM “hash” is attacked first
- Considerations
  - Very slow
  - All passwords will eventually be found
  - Attack against NT hash is MUCH harder than LM hash

Offline Attacks

Pre-computed Hashes

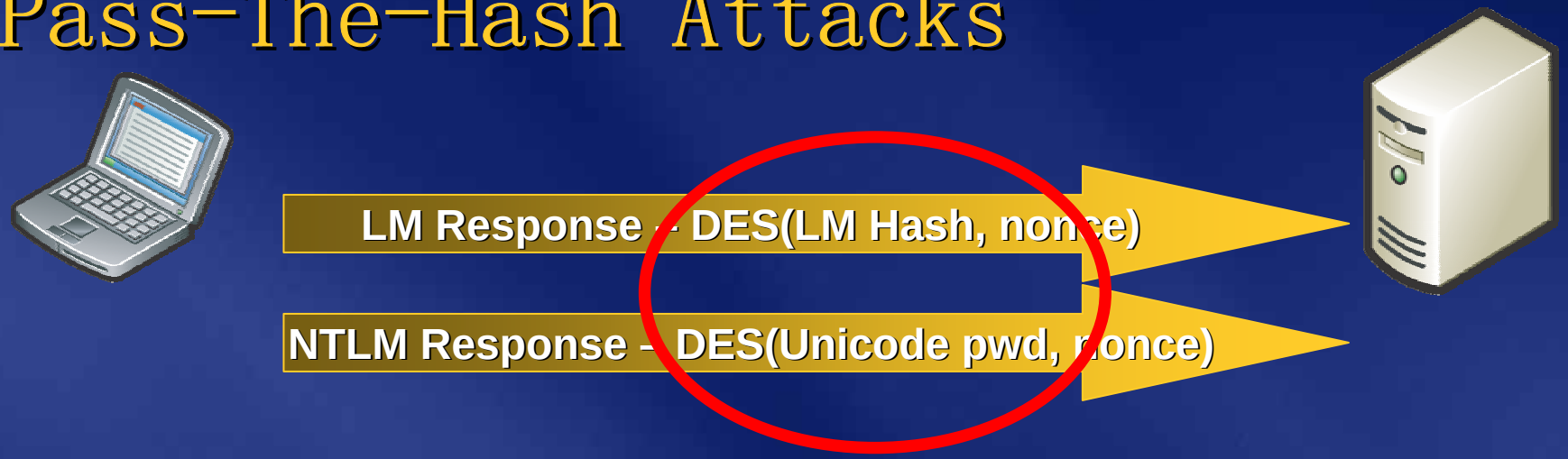
- Generate all possible hashes a priori
- Compare to database values
- Storing hashes requires huge storage
  - LM “Hashes”: 310 Terabytes
  - NT Hashes < 15 chars: 5,652,897,009 exabytes
- Solution: Use a time-space tradeoff
- Succeeds due to lack of salt

Offline Attacks

Pre-computed Hashes – Considerations

- Takes significant effort up front
- LM Hashes much more vulnerable due to smaller key space and shorter length
- Web services available
- SETI-style efforts to generate tables
- Do not work against cached credentials
- Mitigations
  - Use good passwords
  - Remove LM Hashes

Pass-The-Hash Attacks



- Tool computes response from nonce based on arbitrary hash
- Tools are rare but are available
- Instant attack
- Does not work with cached credentials

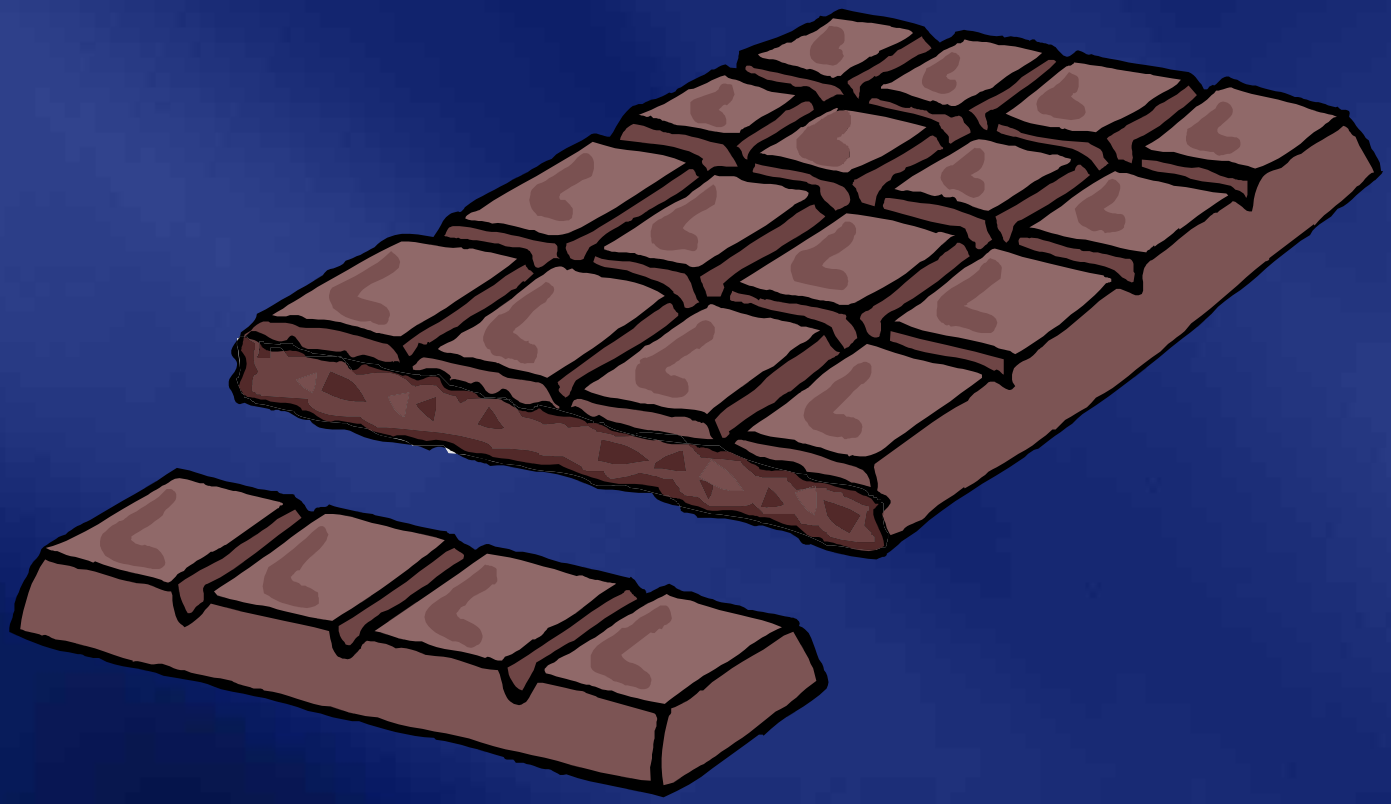
demo

Offline Dictionary And Brute Force Attack

Non-Technical Attacks

- Shoulder surfing
  - Watching someone type their password
  - Common and successful
  - Mouthing password while typing
- Keyboard sniffing
  - Hardware is cheap and hard to detect
  - Software is cheap and hard to detect
  - Both can be controlled remotely
- Social engineering…

Password Cracking at Layer 8



http://zdnet.com.com/2100-1105_2-5195282.html

http://story.news.yahoo.com/news?tmpl=story&cid=528&e=1&u=/ap/20050317/ap_on_go_ca_st_pe/irs_computer_security

Great Password, Weak Implementation

Password Best Practices

Pass Phrases v. Passwords

Longer Is Better!

Technology-Based Mitigation

- Disable LM hash storage
  - HKLM\SYSTEM\CurrentControlSet\Control\Lsa\NoLMHash
  - Passwords > 14 characters
  - Certain Unicode characters
  - Clustering, Windows CE, RTC, ??? broken
    - Set NtlmMinClientSec & 0x80010
- Deploy password policy
  - Minimum length
  - Complexity
  - Expiration
  - Reuse

Password Filter

```
if(strInList(szPwd,aBadWords))
    bComplex = FALSE;

if(cchPassword > 9){
    for(i = 0; i < cchPassword ; i++){
        if(szPwd[i] & C1_DIGIT) { dwNum = 1; continue; }
        if(szPwd[i] & C1_UPPER) { dwUpper = 1; continue; }
        if(szPwd[i] & C1_LOWER) { dwLower = 1; continue; }
        if(szPwd[i] & C1_SYMBOL) { dwSym = 1; continue}
        if(isUnicode(szPwd[i])) {dwUnicode = 1; continue}
}

if(bUserIsAdmin){ //Admins need better passwords than users
    if ((dwNum + dwUpper + dwLower + dwSym + dwUnicode == 5) && cchPassword>14)
        bComplex = TRUE;
}
else { //User is not an admin, use lower requirements
    if(dwNum + dwUpper + dwLower + dwSym + dwUnicode) >= 4)
        bComplex = TRUE;
}
```

Technology-Based Mitigation

Multi-factor authentication

- Why use passwords at all?
- Smart cards
  - Two-factor authentication
  - Very difficult to thwart
  - High cost of initial deployment
- Biometric
  - Two- or three-factor authentication
  - Usually defeated with non-technical attacks
  - Very expensive
  - Failure-prone

Fun With Biometrics

Detecting Attacks – Account Lockout

Summary

- How passwords are stored
- How passwords are used
- How passwords are attacked
- Password best practices

Passwords are like

bubblegum

Strongest when fresh

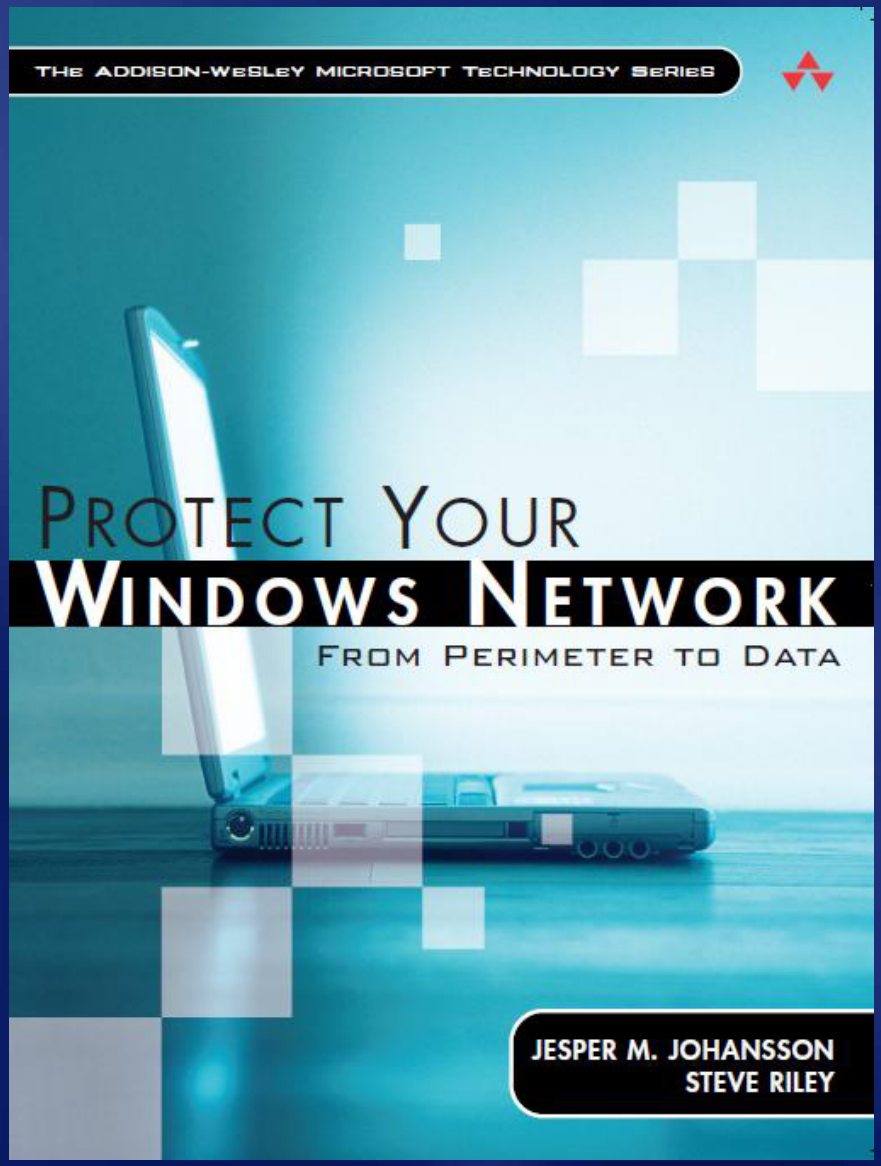
Should be used by an individual, not a group

If left laying around, will create a sticky mess

Passwords Article Series

http://www.microsoft.com/technet/security/secnews/newsletter.htm

For more information



Jesper and Steve finally wrote a book!

Order online:
http://www.awprofessional.com/title/0321336437

Use promo code
JJSR6437

jesperjo@microsoft.com

Jesper M. Johansson
jesperjo@microsoft.com

Microsoft®

Your potential. Our passion.™