CS 465

PASSWORDS

Slides by Kent Seamons and Tim van der Horst
Last Updated: Oct 25, 2016

Goals

- Understand UNIX pw system
  - How it works
  - How to attack
- Understand Lamport's hash and its vulnerabilities

History of UNIX passwords

- Originally the actual passwords were stored in a plaintext file
  - “Excessively vulnerable to lapses in security”
- Improved approach used encryption to protect passwords
  - Led to brute force/dictionary attacks

Pass Phrases

- Passwords is a misnomer
  - Do not use single words or variants
  - Supposedly, a large number of passwords in Dallas is some variant of the word cowboys
    - Any cougar passwords out there!
- Use a pass-phrase
  - Memorable and harder to guess
  - First letter of a long phrase
    - Rastcao - Rise and shout the cougars are out

How to Attack Password Systems

- Guess the user's password
  - Online attack
    - Attempt to login as the user would
  - Offline attack
    - Repeated guessing involving an encrypted form of the user's password
- Shoulder surfing
- Users write down their passwords
- Users give away their passwords
  - Phishing, social engineering

Problems with Passwords

- Users have too many passwords
  - Encourages password reuse
  - Leads to forgotten passwords
  - Burdens users and administrators
- Attempts to increase password strength inconvenience users
- Random passwords
  - Only as random as the initialization of the salt value

Time estimates

- What is the maximum number of attempts to guess a password?
  - Password length = 8 characters
  - Assume password is alphanumeric (26+26+10)
  - $(26+26+10)^8 = 62^8$
- How many attempts on average? Divide maximum number by 2 (this assumes brute force attack and passwords chosen randomly)

Unix Passwords

Unix Password File

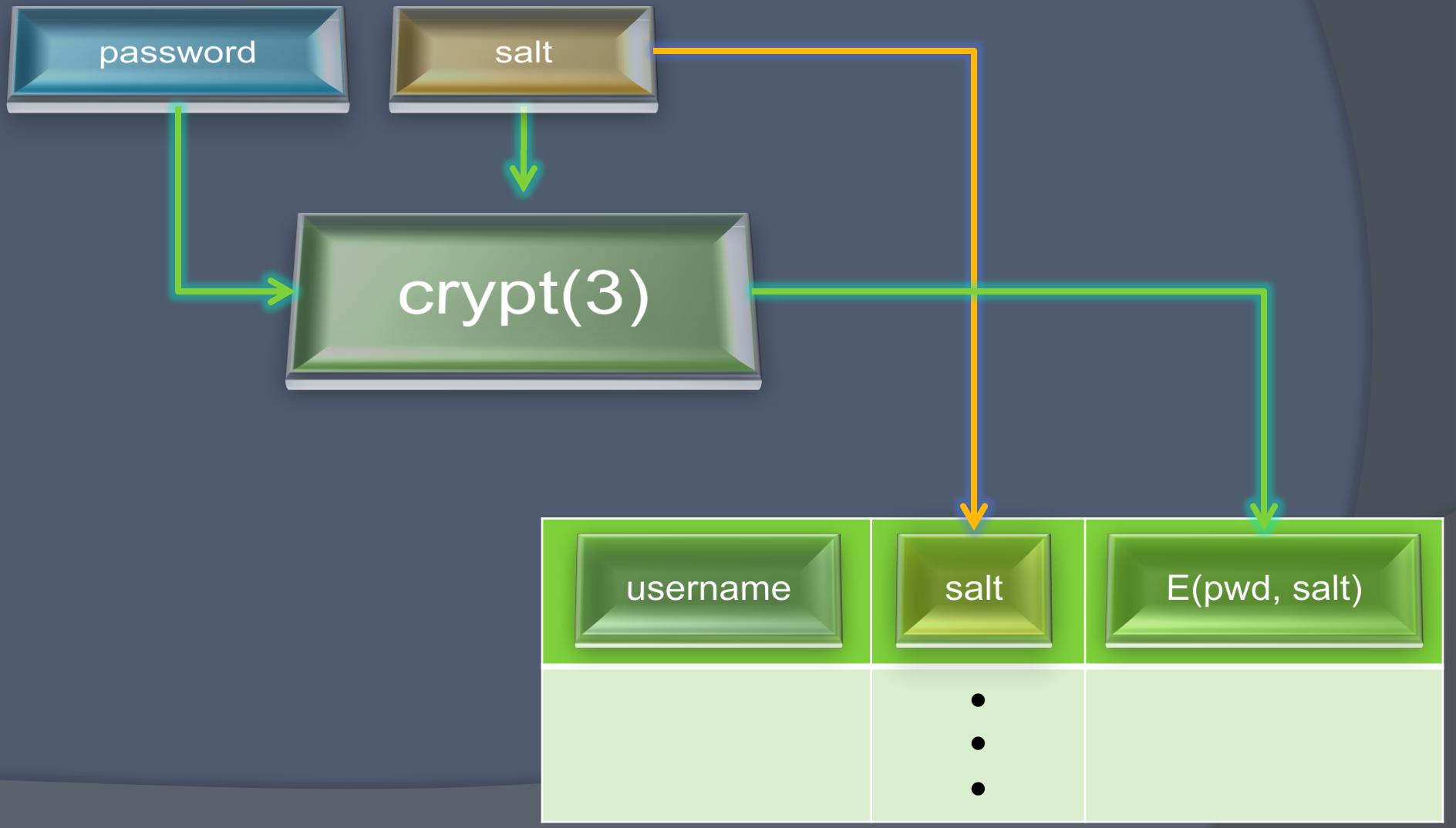
- Original password file /etc/passwd was world readable
  - Anyone could copy the file offline and perform a dictionary attack
  - You could find sample files on Google courtesy of naïve system admins!
- Later, the encrypted password was moved to a shadow file /etc/shadow that required root privileges to access

THE UNIX CRYPT FUNCTION

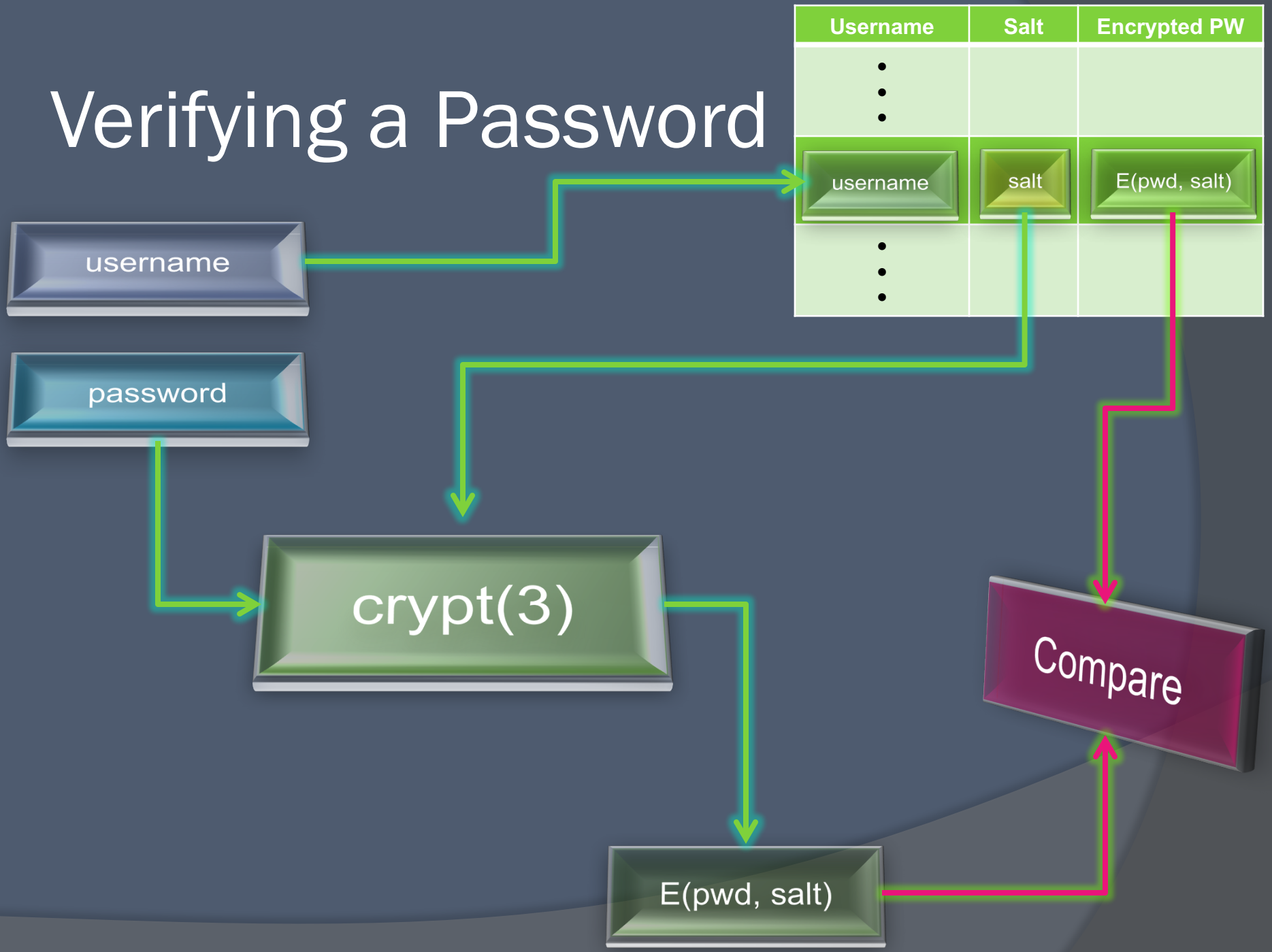
crypt(3)

Slower is better

Unix Password File Creation



Verifying a Password



Password Salts

- Why do Unix password files use a salt?
  - Prevents the identification of identical passwords
    - Provided each user has a different salt
  - All password guesses are salt-specific
    - Guess made with one salt aren't helpful for another
    - Increases the cost of offline attack to crack any password in the file
    - Increases the size requirement for a pre-computed database of hashed passwords

Password Attacks with Salt

- How many guesses do password attacks need when a salt is used?
  - Off-line attack – one attempt for each unique salt in the file
- How does the salt impact on-line attacks?
  - It doesn't
- How does the salt impact an attempt to crack a specific user's password in the file?
  - It doesn't change the number of attempts, but it does increase the size of a pre-computed database of passwords or rainbow table

Password Hashing Schemes

Scheme id	Schema	Example
	DES	`Kyq4bCxAXJkbg`
_	BSDi	`_EQ0.jzhSVeUyoSqLupI`
1	MD5	`$1$etNnh7FA$OlM7eljE/B7F1J4XYNnk81`
2, 2a, 2x, 2y	bcrypt	`$2a$10$VIhIOofSMqgdGlL4wzE//e.77dAQGqntF/1dT7bqCrVtquInWy2qi`
3	NTHASH	`$3$$8846f7eaee8fb117ad06bdd830b7586c`
5	SHA-256	`$5$9ks3nNEqv31FX.F$gdEoLFsCRsn/WRN3wxUnzfeZLoooVlzeF4WjLomTRFD`
6	SHA-512	`$6$qoE2letU$wWPRl.PVczjzeMVgjiA8LLy2nOyZbf7Amj3qLIL978o18gbMySdKZ7uepq9tmMQXxyTIrS12Pln.2Q/6Xscao0`
md5	Solaris MD5	`$md5,rounds=5000$GUBv0xjJ$$mSwgIswdjlTY0YxV7HBVm0`
sha1	PBKDF1 with SHA-1	`$sha1$40000$jtNX3nZ2$hBNaIXkt4wBI2o5rsi8KejSjNqIq`

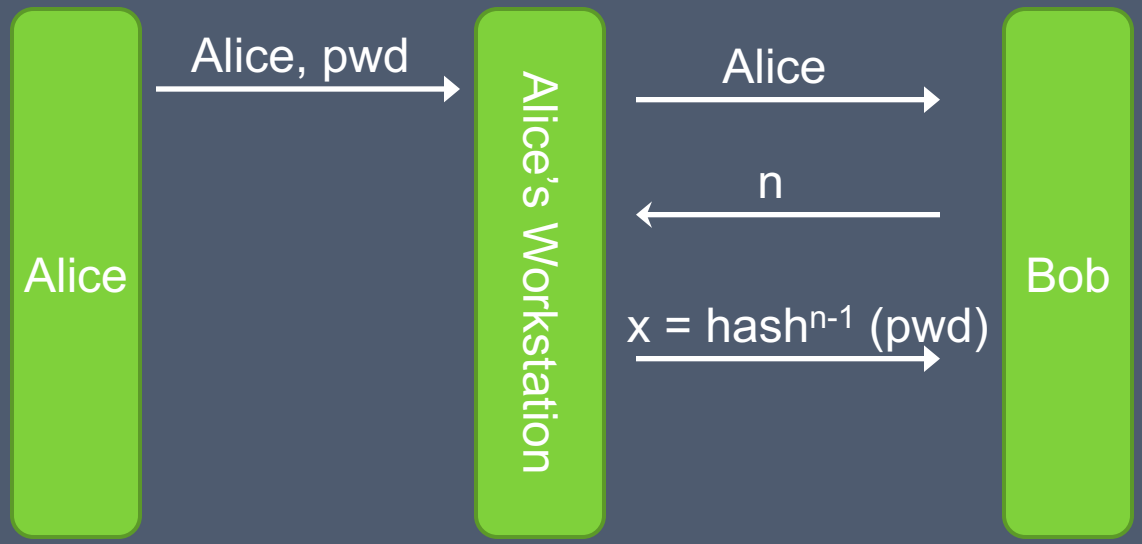
Password Guessing Attacks

- Brute-force
- Dictionary
- Substitution
  - password, passw0rd

Lamport's Hash

Lamport's Hash

- One time password scheme
  see http://lodestone.org/people/hoss/ops/node5.html



BOB

knows ⟨ n, hash^n(password) ⟩

Compares hash(x) to
hash^n (password);

If equal, replaces
⟨ hash^n (password) ⟩
With
⟨ n-1, x ⟩

Attack on Lamport's Hash

- Small n attack
  - Active attacker intercepts servers reply message with n and changes it to a smaller value
  - Attacker can easily manipulate the response (repeatedly) to impersonate Alice
- Eavesdropper captures Alice's hashed reply and conducts off-line attack
- Replay Alice's response to other servers where Alice may use the same password
  - Thwart using salt at the server – server hashes pw || salt and sends n and the salt to Alice during login
  - Salt also permits automatic password refresh when n reaches 1

Related articles (optional)

- The Curse of the Secret Question
  http://www.schneier.com/essay-081.html
- Sarah Palin Yahoo! account hacked
  http://www.informationweek.com/news/security/cybercrime/showArticle.jhtml?articleID=210602271
  http://en.wikipedia.org/wiki/Sarah_Palin_email_hack
- Secret Questions Too Easily Answered
  http://www.technologyreview.com/web/22662/
- Scientists claim GPUs make passwords worthless
  http://www.pcpro.co.uk/news/security/360313/scientists-claim-gpus-make-passwords-worthless
- How the Bible and Youtube are fueling the next frontier of password cracking
  http://arstechnica.com/security/2013/10/how-the-bible-and-youtube-are-fueling-the-next-frontier-of-password-cracking/
- 32 million passwords show most users careless about security
  http://arstechnica.com/security/2010/01/32-million-passwords-show-most-users-careless-about-security/