پروتکلهای احرازاصالت

Authentication protocols

فهرست مطالب

مقدمه

- چرا احراز اصالت؟
  - حضور موجودیتهای گوناگون و پرشمار در شبکه
  - داده‌ها و سرویسهای مختلف باید در اختیار موجودیتهای خاص و شناخته‌شده قرار گیرند.
    - هر موجودیتی حق دسترسی به هر سرویس و داده‌ای را ندارد.
  - برخی موجودیتها ممکن است هویت برخی دیگر را جعل کنند.
    - دسترسی به سرویسها و اطلاعات حساس، پنهان کردن هویت موجودیت خرابکار، فرار از حسابرسی و ... .

مقدمه

- احراز اصالت موجودیتها: تأیید هویت مورد ادعای یک موجودیت در زمان معین مورد نظر
  - احراز اصالت یکطرفه
  - احراز اصالت دوطرفه
- عملیات احراز اصالت معمولاً با انجام تعدادی تبادلات رمزنگاری انجام می‌شود که پروتکل احراز اصالت نام دارد.

Basis for Authentication

- Something you **know** (a PIN, or password).
- Something you **have:**
  - secureID card or other token, generating a one-time password.
  - a key imbedded in a ‘secure area’
  - a smartcard (which may have keys imbedded and can perform cryptographic operations on behalf of a user).
- Something you **are** (a biometric).

فهرست مطالب

- مقدمه
- احرازاصالت ضعیف(کلمات عبور)
- احرازاصالت قوی (مبتنی بر سؤال و جواب)
- پروتکل کربروس

احراز اصالت ضعیف

- احراز اصالت ضعیف
- احراز اصالت یک طرفه
- انتخاب یک رشته از کاراکترها توسط کاربر به عنوان راز مشترک بین کاربر و سیستم
- وارد کردن کلمه عبور در هر بار احراز اصالت

احراز اصالت ضعیف

- **فایل حاوی متن ساده کلمات عبور**
  - ذخیره متن ساده کلمات عبور در سیستم
  - محافظت از فایل در برابر خواندن و نوشتن
  - غیر مبتنی بر رمزنگاری
  - ارسال فاش کلمه عبور بر روی خط ارتباطی کاربر و سیستم
  - امکان حمله تکرار
  - امکان سوء استفاده افراد دارای حق دسترسی

احراز اصالت ضعیف

- **فایل حاوی مقدار درهم کلمات عبور**
  - درهم سازی کلمات عبور و ذخیره نتیجه در سیستم
  - محاسبه مقدار درهم کلمه عبور وارد شده و ارسال آن بر روی خط ارتباطی کاربر و سیستم
  - محافظت از فایل فقط در برابر نوشتن
  - امکان حمله تکرار

Password Vulnerabilities

- Writing them down

- Stolen passwords (via eavesdropping)
  - Trojan Horse
- Poor password choice
  - Easy to guess, easy to remember
  - People use the same password multiple times
  - Passwords changed infrequently
- Offline attacks
  - Search through password dictionary

Survey of 3,289 Passwords

- With no constraints on choice of password
  - 15 were a single ASCII letter.
  - 72 were strings of two ASCII letters.
  - 464 were strings of three ASCII letters.
  - 47 were strings of four alphanumerics.
  - 706 were five letters, all upper-case or all lower-case.
  - 605 were six letters, all lower case.

حملات علیه روشه های مبتنی بر کلمه عبور و راه کار مقابله

- جستجوی کامل
- جستجوی برخط. روش های مقابله:
  - اطمینان از انتخاب کلمات عبور از بین یک فضای بزرگ
  - محدودیت گذاشتن روی تعداد دفعات وارد کردن نا موفق کلمه عبور در یک بازه زمانی معین
  - کاهش سرعت فرآیند نگاشت و یا وارد کردن کلمه عبور
- جستجوی برون خط
  - استفاده از فایل حاوی مقدار درهم کلمات عبور
  - وابستگی موفقیت حمله به تعداد کلمات عبور، نوع نگاشت آنها و میزان پردازنده در اختیار (برای موازی سازی)

حملات علیه روشه های مبتنی بر کلمه عبور و راه کار مقابله

- حمله تکرار
  - شنود خط ارتباطی
  - مشاهده هنگام تایپ یا احتمالا جایی که نوشته شده
- حملات حدس کلمه عبور و واژه نامه
  - استفاده از واژه نامه های مرسوم، برخط و یا تخصصی
  - انتخاب کلمه عبور از یک فضای بسیار کوچک

بهبود امنیت در روش های مبتنی بر کلمه عبور

- قوانین کلمه عبور
- کاهش سرعت نگاشت کلمه عبور
- نمک زدن به کلمات عبور
- استفاده از عبارات عبور
- سازوکارهای بازدارنده

نمک زدن

- افزودن یک رشته تصادفی t بیتی (نمک) به کلمه عبور و سپس اعمال تابع درهم ساز بر آن
- ذخیره فاش نمک به همراه مقدار درهم کلمه عبور
- دشوار کردن حملات واژه نامه
- عدم تاثیر بر عملکرد حدس کلمه عبور یک کاربر مشخص

عبارتهای کلمه عبور

- وارد کردن یک عبارت به جای یک کلمه
- افزایش آنتروپی با زیاد کردن تعداد کاراکترها
  - آنتروپی یک کاراکتر در یک کلمه یا عبارت عبور در مقابل یک رشته تصادفی
- نیاز به تایپ اضافی
  - وقت گیر بودن
  - افزایش احتمال اشتباهات تایپی

احراز اصالت دو عاملی

- شماره شناسایی شخصی
- استفاده به همراه یک توکن
  - وارد کردن PIN هنگام استفاده از توکن
  - ایجاد سطح دوم امنیت هنگام گم یا دزدیده شدن توکن
- کوتاه و شماره ای
  - ۴ تا ۸ رقم
  - امکان جستجوی کامل
- توقیف و یا غیرفعال کردن توکن در صورت چند بار اشتباه وارد کردن PIN

کلمه عبورهای یک بار مصرف

- فقط یکبار استفاده از هر کلمه عبور
- مقاوم در برابر شنود به منظور جعل هویت بعدی
- سه روش اصلی پیاده سازی
  - استفاده از لیست های از پیش مشترک کلمات عبور
    - نگهداری امن لیست در دو طرف
    - حفظ ترتیب
  - استفاده از کلمات عبوری که متناوباً به روز می شوند
    - توافق روی کلمه عبور بعدی در هر ارتباط
    - ایجاد مشکل در صورت قطع ارتباط

کلمه عبورهای یک بار مصرف

- دنباله های کلمه عبور یکبار مصرف مبتنی بر توابع یک طرفه
  - کاراتر از دو روش قبل
  - نوعی پروتکل چالش- پاسخ با تعیین غیرمستقیم چالش توسط موقعیت جاری در دنباله
  - مثال: شمای کلمه عبور یکبار مصرف Lamport

کلمه عبورهای یک بار مصرف

- انتخاب راز w، ثابت t و ارسال $w_0=H^t(w)$ برای سیستم توسط کاربر
- مقداردهی $i_A=1$ توسط سیستم
- iامین بار احراز اصالت $A \rightarrow B: A, i, w_i \ (= H^{t-i}(w))$
- بررسی $i=i_A$ و $H(w_i)=w_{i-1}$ توسط سیستم
- افزایش شمارنده و ذخیره w_i برای نشست بعدی

W
H(-)

$\{A, i, H^{(t-i)}(w)\}$

$H^{(t-i+1)}(w)$,
H(-)

کلمه عبورهای یک بار مصرف

- انتخاب راز اولیه p توسط کاربر و ارسال آن برای کارگزار
- ارسال (r, H(r, p)) در هر مرحله احراز اصالت
  - r یک مقدار تصادفی، مهر زمانی و یا شماره ترتیب
- امکان جعل هویت به وسیله دستبرد به اطلاعات کارگزار
  - ضعف این روش نسبت به پروتکل Lamport

کلمه عبورهای یک بار مصرف

- ضعف در برابر دشمن فعالی که پیام کاربر را سد کرده و بعداً برای کارگزار ارسال می کند
- همچنان امکان اجرای حمله حدس کلمه عبور با استفاده از پیام های مبادله شده
- راه حل: استفاده از پروتکل های چالش- پاسخ مبتنی بر کلمه عبور

فهرست مطالب

- مقدمه
- احرازاصالت ضعیف(کلمات عبور)
- احرازاصالت قوی (مبتنی بر سؤال و جواب)
- پروتکل کربروس

Strong Authentication

- In **strong** authentication, one entity 'proves' its identity to another by demonstrating knowledge of a secret known to be associated with that entity, *without revealing that secret itself during the protocol.*
- Also called `challenge-response' authentication.
- Use cryptographic mechanisms to protect messages in protocol:
  - Encryption.
  - Integrity mechanism (e.g. MAC).
  - Digital signature.

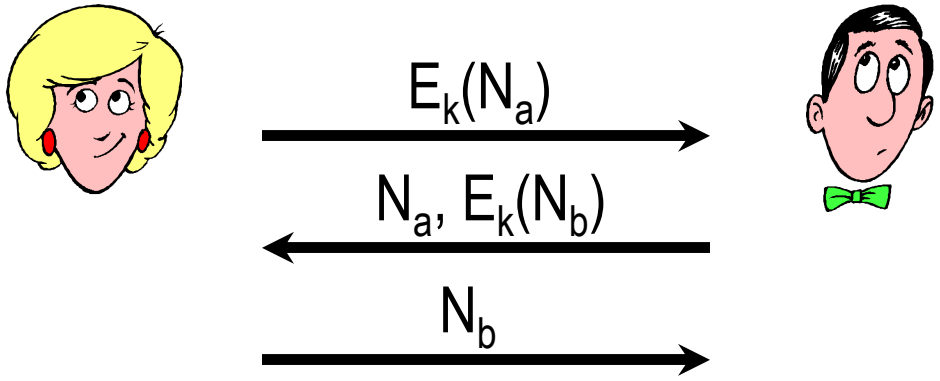
Encryption-based Unilateral Authentication

1. A → B: 'Hi Bob, I'm Alice'
2. B → A: R (challenge)
3. A → B: $\{R \| B\}_K$ (response)

(Here, $\{X\}_K$ means string X encrypted under key K, and $\|$ means concatenation of strings.)

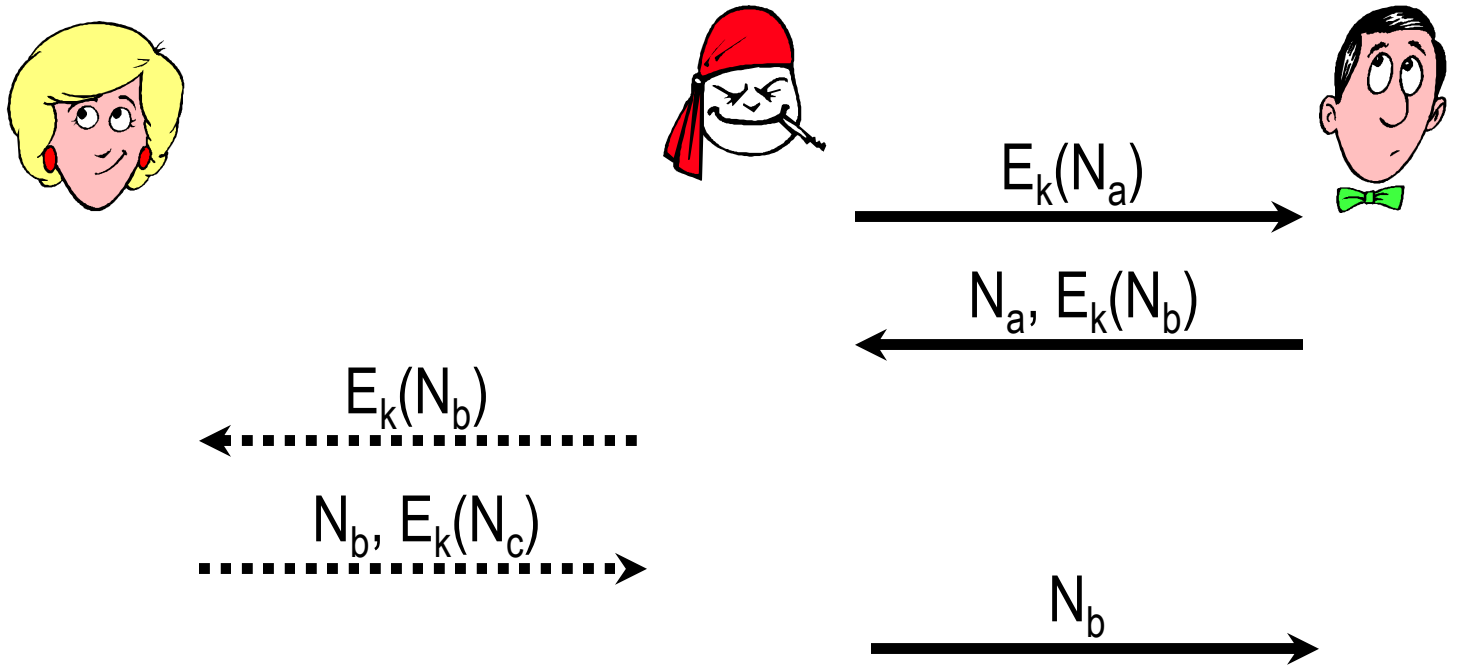
Standard Authentication

- Alice and Bob share (strong) key k
- Simple challenge-response type protocol: (N_a,N_b - nonces)



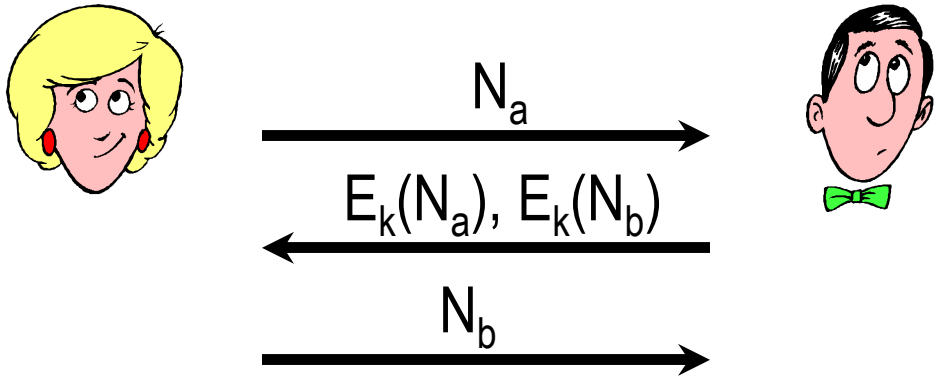
Attack on Simple Protocol

- "Oracle Attack"



Protocol Fix?

- Modified challenge-response type protocol:



این پروتکل نیز آسیب پذیر است

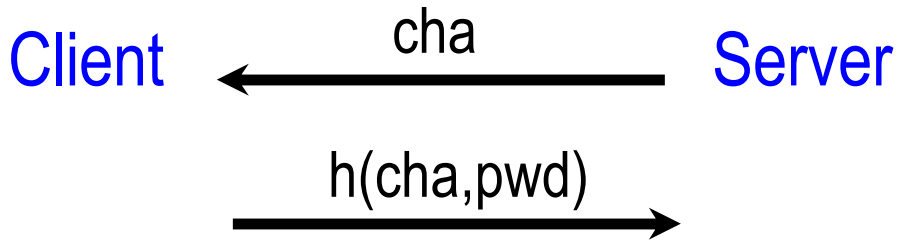
Password-based Protocols

- Telnet - vulnerable to replay attacks

- Hashing does not help

Password-based Protocols

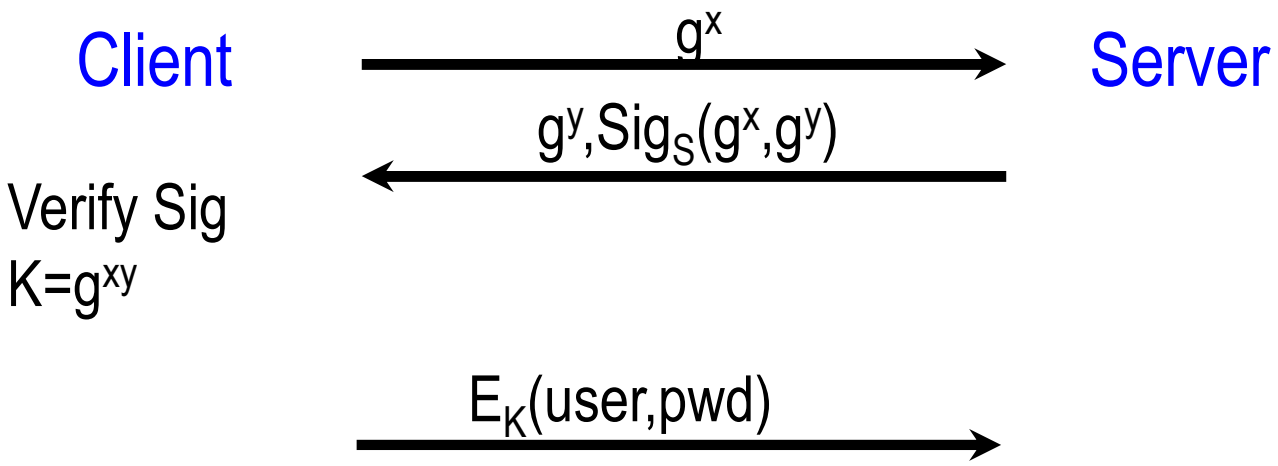
- Challenge-Response : vulnerable to offline dictionary attacks



- Problem: “verifiable text”

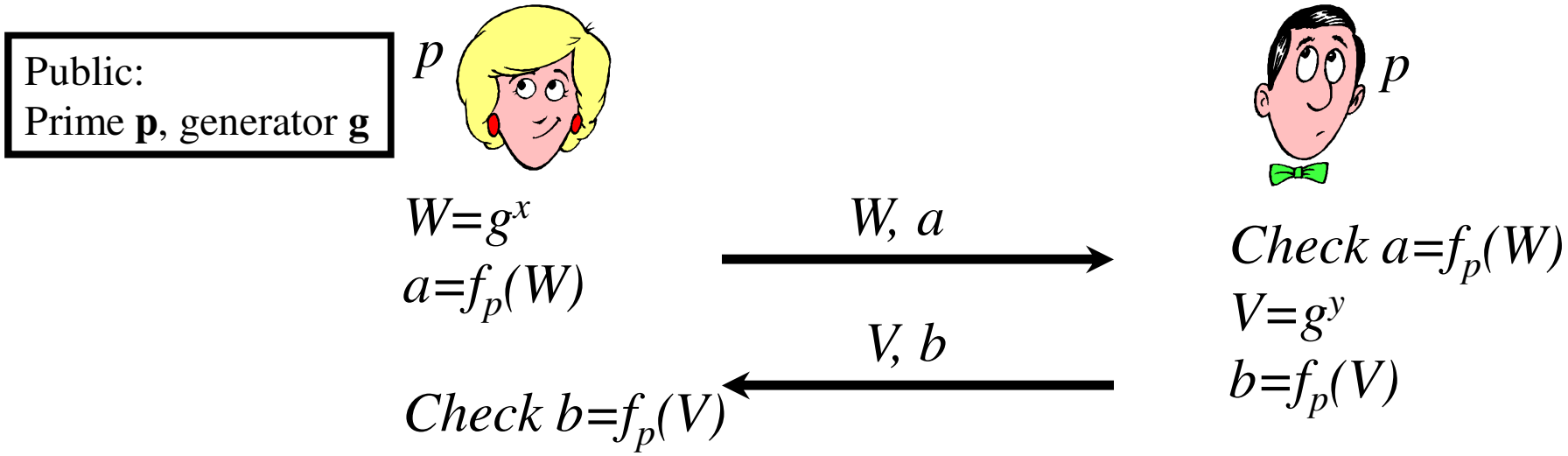
Password-based Protocols

- SSH: Relies on public key



- Similar protocols relying on public keys:
  - [SSL],[Halevi-Krawczyk],[Boyarsky],[Shoup]

Password Auth. - Attempt



- **Intuition:** authenticate Diffie-Hellman values using PRF with password as key.
- **Insecure!** (eavesdropper obtains verifiable text)