Challenge – Response

A fundamental problem with passwords is that passwords are used repeatedly. When a password is intercepted, the authentication system cannot determine if the real user is entering the password or if an imposter is supplying the password. One strategy to counter this situation is to allow a password to be used only once. There are several different types of challenge-response systems.

Def: Let user *U* desire to authenticate himself to system *S*. Let *U* and *S* have an agreed-on secret function *f*. A *challenge-response* authentication system is one in which *S* sends a random message *m*, the challenge, to *U*. *U* replies with the transformation *r* = *f*(*m*), the response. *S* validates *r* by computing *r* separately.

Note:

This technique is similar to the IFF (identification – friend or foe) that the military uses to identify allied and enemy planes.

Pass Algorithms

e.g.

Suppose that we have a challenge-response system where the agreed upon transformation is the mapping below:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7	8	9

In this mapping a random string of lower-case letters is generated and presented to the user for authentication. The user has the above mapping and translates the lower-case letters in the upper line into the corresponding characters in the lower line. The user then presents those characters to the system for authentication. If the challenge is

```
h   a   r   o   l   d
```

The response would be

```
7   0   1   E   B   3
```

Def: In a challenge-response authentication system in which the function *f* is a secret, the function *f* is called a *pass algorithm*.

One-Time Passwords

The ultimate form of password aging occurs when a password is valid for exactly one usage.

Def: A *one-time* password is a password that is invalidated as soon as it is used.

The challenge is the number associated with the authentication attempt. The response is the one-time password for that authentication prompt.

Some of the problems associated with this approach are the generation of random passwords, the distribution of those passwords, and the synchronization of the user and the authentication system. A cryptographic hash function or enciphering function such as DES could be used to generate the one-time passwords. The issue relative to the distribution of the passwords has traditionally been solved by using a secure courier to deliver the passwords. Other secure delivery techniques could also be used. Public key encryption techniques could be used to distribute the passwords electronically.

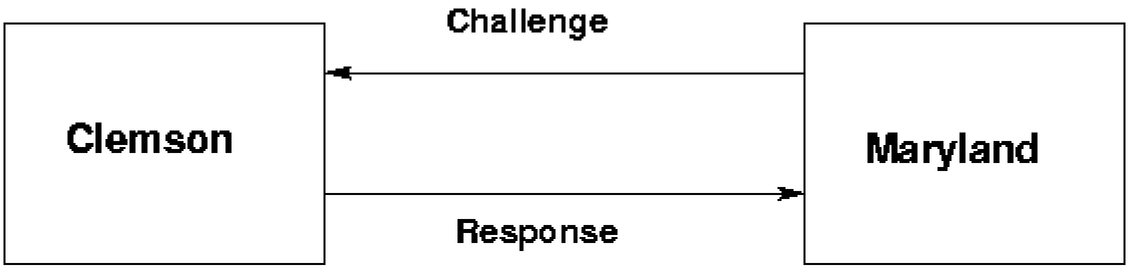
The last issue is the synchronization of the user and the authentication system. This can be accomplished by numbering the passwords. In this case, the authentication system would prompt the user for password n. The user would look on the supplied list and respond with password n. The authentication would then delete that password from the list. When the list is empty, a new list of one-time password would need to be generated and sent to the user.

Hardware-Supported Challenge-Response Procedures

Hardware support comes in two forms:

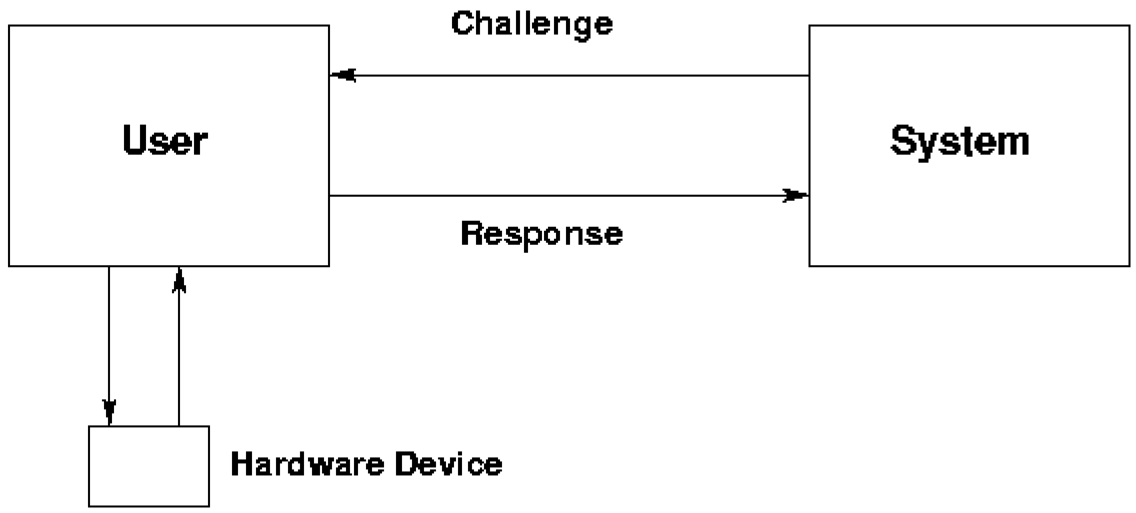
a program for a general purpose computer
special-purpose hardware support

Software



Talk about Darrell and Kim.

Hardware



Process:

User provides login name
System sends challenge
User enters challenge into device
Device responds with an appropriate response
User enters response & password
System validates response & password

Challenge-Response & Type 1 Dictionary Attacks

A dictionary type of attack is possible with a challenge-response system if the attacker knows the challenge and response.

Suppose that Oscar is listening and knows

1. challenge
2. response

Same type of attack as for a re-usable password

If Oscar sees enough challenges & responses, then he might guess challenge-response algorithm.