ICS 351: Today's plan

- HTTPS: SSL and TLS
- certificates
- cookies
- DNS reminder
- Simple Network Management Protocol
secure HTTP

- HTTP by itself is very insecure: any man-in-the-middle attacker can observe all the content sent and received
- some people wish to use HTTP to send sensitive data, e.g. credit card numbers, personal email
- instead of layering HTTP over TCP, HTTP can be layered over a secure protocol that runs over TCP
- the choice of secure protocols for HTTPS (secure HTTP) is SSL (older) or TLS (newer)
- both SSL and TLS are considered secure, but
- SSL and TLS authentication requires a public key for the server
- how to connect to a server that has not been visited before?
certificates

- a *certificate* is a digital signature by entity CA verifying that the enclosed public key authenticates server S
- there are a few (~100) certificate authorities (CAs) that are widely known and recognized by many web browsers
- when presenting its public key, a server S also presents the certificate signed by a CA as evidence that S indeed is the server the user wants to talk with
certificate vulnerabilities

- certificates protect against man-in-the-middle attack (including DNS attacks), but are still vulnerable to misspellings (e.g. goggle.com)
- if the certificate authority is compromised, and DNS or the routing infrastructure subverted, an attacker can impersonate any website
- this may have happened – the dutch CA diginotar may have had its keys stolen and misused
self-signed certificates

• if I have a website for private use, I don’t need a certificate from a CA
• I can use a self-signed certificate instead
• as before, the crucial step is giving the browser the correct public key for the desired server
• this requires hand-configuration of all the browsers that will use this server
HTTP cookies

- HTTP is a stateless protocol: a server has no real way to identify a client, so a request may or may not be connected with prior requests
- instead, a server may offer a client a *cookie*, a small amount of data that is only meaningful to the server
- on subsequent related requests to the same server, the client will send back the cookie, to confirm that the requests are connected
- cookies have an expiration time -- most cookies used for authentication expire quickly
HTTP cookies

• cookies can also be used to attempt to track users as they visit multiple sites, by embedding in the several sites a small image (or other content) served from the same server

• these cookies are often long-lived

• similar tracking can be done by tracking accesses based on the IP number of the connecting client
DNS reminder

- DNS provides name to IP address resolution
- Domain names are grouped into zones
- a DNS server provides translation (resolution) for the names in one zone
- a DNS query contains *question* Resource Records
- a DNS response may contain *answer* RRs, *name server* RRs, and *additional* RRs
dig hawaii.edu

;; QUESTION SECTION:
;hawaii.edu. IN A

;; ANSWER SECTION:
hawaii.edu. 1800 IN A 128.171.224.100

;; AUTHORITY SECTION:
hawaii.edu. 1800 IN NS dns4.hawaii.edu.
hawaii.edu. 1800 IN NS dns2.hawaii.edu.
hawaii.edu. 1800 IN NS dns1.hawaii.edu.

;; ADDITIONAL SECTION:
dns1.hawaii.edu. 1800 IN A 128.171.3.13
dns1.hawaii.edu. 1800 IN A 128.171.1.1
dns2.hawaii.edu. 1800 IN A 128.171.3.13
dns2.hawaii.edu. 1800 IN A 128.171.1.1
dns4.hawaii.edu. 1800 IN A 130.253.102.4
dig mx hawaii.edu

;; QUESTION SECTION:
;hawaii.edu. IN MX

;; ANSWER SECTION:
hawaii.edu. 1800 IN MX 10
mx1.hawaii.edu.

;; AUTHORITY SECTION:
hawaii.edu. 1800 IN NS dns1.hawaii.edu.
hawaii.edu. 1800 IN NS dns4.hawaii.edu.
hawaii.edu. 1800 IN NS dns2.hawaii.edu.
system administration

- suppose a system administrator has to manage a large number of machines
- for example, three web servers, a DHCP server, a backup server, a Network Attached Storage (NAS) server, a mail server, and a few printers
- a large KVM might be useful, but also has limitations:
  - all the servers must be in close physical proximity
  - there cannot be multiple, remote consoles
  - there is no way to get alerts from systems that need attention
Simple Network Management Protocol

- SNMP uses the network to report status information and alerts about remote systems.
- SNMP messages are carried over UDP.
- Values can be loaded on demand (pull model), but when needed and configured appropriately, alerts are sent independently by the systems being managed (push).
SNMP Management Information Base

- SNMP needs a machine-independent way to indicate which item of information is being requested or sent
- Logically, the entire universe of information that can be accessed is built into a large tree: the Management Information Base or MIB
- The tree is extensible so individuals and organization can add their own subtrees -- private MIBs
- The tree is universal and known to all
navigating the MIB

- the path through the tree is sufficient to indicate one specific item (corresponding to a variable in a programming language)
- the path through the tree can be indicated by a sequence of numbers, the number of left siblings of the path being taken
- for example, 0.2.7.5.14.1.7.0 is such an Object Identifier (OID)
- OIDs are useful for enumerating arrays of objects, e.g., network interfaces, routing table entries
SNMP programs

- a network management station is used by the system administrator to monitor multiple systems
- a management agent must run on every managed device, get the required information, and provide it on request
SNMP basic operation

- The network management station may send GET requests to get one or more objects from specific agents.
- The network management station may also send SET requests to modify one or more objects on specific agents.
- Agents will send TRAP or INFORM alerts to network management stations that they have been configured to alert.
- Because it uses UDP, SNMP (like DNS) cannot assume that its operations will be successful.