

## HW #8: Threat Modeling

CS 392/6813: Computer Security  
Fall 2007

**[100pts] DUE 11/27/2007 (midnight)**

### **Problem 1 [60pts]**

Tor is a system that provides anonymity to users on the internet (we will be studying about this in coming lecture). It does so by re-routing user requests via a series of three routers in such a manner that the recipient can not figure out the IP address of the sender. The first router knows the IP of sender but does not know the IP of destination, the last router knows the IP of destination but not the IP of source. Basically, the sender sends out a request  $m$  as a triple encryption  $\text{Enc}(K1, \text{Enc}(K2, \text{Enc}(K3, m, D), R3), R2)$ , where  $R1-R2-R3$  are the three routers,  $K1, K2, K3$  are the public keys of  $R1, R2, R3$ , respectively, and  $D$  is the destination. Each router simply chops of the outer layer of encryption, retrieves the address of the next router/destination  $X$  and forwards the inner layer to  $X$ . Unless all three routers collude with each other, it is “hard” to figure out the link between the sender and the destination, i.e., who communicated with whom.

In this problem, your task is to perform the threat modeling for Tor, i.e., you have to build an attack tree that shows all possible ways to **break the privacy** provided by the Tor system, i.e. to figure out who communicated with whom. It will be better to use a “textual representation”. The Tor website is here <http://www.torproject.org/>. To find out about various privacy attacks on Tor, you will have to do some research. Here is a very recent paper on this: <http://www-users.cs.umn.edu/~hopper/ccs-latency-leak.pdf>.

### **Problem 2 [40pts]**

Image-based spam is a new class of spam successfully used by spammers these days. Such spam contains its unwanted content inside of graphics (such as in html embedded images), making it difficult for text-based spam filters to identify. In an organization, “Yet\_Another\_Finalcial\_Firm\_in\_NY”, there are 100 employees. Each employee receives on an average 40 emails per day. Every day, one-third of all emails received are image-based spam emails. To transmit each image-based email (of average size 200bytes) from the mail server to an employee’s mailbox, there is an extra bandwidth overhead of 200 bytes.

Researchers at Poly have developed a new open-source (free of cost) software to detect image-based spams. This software has a false positive (i.e., an image-based spam is not

detected) rate of 5%, and it incurs, on average, a computational overhead of 800 milliseconds to process a 200 bytes image-based spam email, when the detection is successful. When the detection fails, the software incurs no computational overhead, though.

Should “Yet\_Another\_Financial\_Firm\_in\_NY” deploy the above software to protect against image-based spam? You can assume that computational cost equivalent to transmit 1 bit (from mail server to mailbox) is 10ms.