CSCI 5271 Exercise Set 5

Alex Biedny

December 13, 2020

Problem 1. Denial of Service Denial

(a) Sly's scheme will not work. A DoS attack works by consuming a limited resource, and Sly's method for preventing Dos attacks still uses several limited resources. First, it requires compute time from the web server to check the delayed bit on every packet, and it would require an $O(n)$ operation to search the queue to check for duplicate sender IP addresses. In addition, as the number of incoming requests grows, putting requests from duplicate sender IPs at the end of the queue means that the queue will need to keep growing, consuming more of a limited resource (the servers memory). If an adversary launches an attack against this server, they would still be able to deny service by consuming all/most of the servers limited compute power and memory, regardless of this scheme.

(b) This would work to some degree, but a skilled adversary would be able to circumvent this scheme easily. If the DoS attacker is repeatedly requesting only the same page from a server, this mitigation mechanism would be effective, as every time the attacked loaded a page, they would be spawning a new server and scaling up the server they're trying to attack. However, this scheme fails in a few ways: First, the attacker is under no obligation to do anything with the server response. This means that a smart attacker could simply send a request to the server and then discard the response, which contained the ActiveX control webserver which will now not run to scale the server. Next, an HTTP redirect is still something that needs to be server, and will consume a small amount of resources on the main server. If the server needs to send a large amount of redirects in a small amount of time, the DoS attack could still work by taxing the servers resources such that it cannot server the HTTP redirects. Finally, an attacker could simple request different pages from the server. Because the ActiveX servers only server one page, by requesting different pages each time, there will be no mini server to redirect to, and the server will have to serve the requested pages, using resources and making the DoS attack work.

Problem 2. Firewall Schmirewall

(a)

SRC ADDR	DEST ADDR	SRC PORT	DST PORT	PROTOCOL	ACTION
10.1.100.100	*	*	sendmail	TCP	ALLOW
*	10.1.100.100	*	sendmail	TCP	ALLOW
*	10.1.100.200	*	ssh	TCP	ALLOW
*	10.1.200.200	*	http/s	TCP	ALLOW
*	0.1.2.3	*	ssh	TCP	ALLOW
*	42.42.42.42	*	ssh	TCP	ALLOW
*	3.14.15.9	*	ssh	TCP	ALLOW
*	*	*	*	*	DENY

(b) These can all be implemented with a firewall at the edge of the network, as all of the rules are applying to all machines in the network. A host based firewall wouldn't be needed; traffic only needs to be restricted at the network edge.

(c) The outgoing SSH connections to client sites could be used to tunnel an unrestrcted internet connection, the current firewall and proxy couldn't stop this without restricting SSH access out of the network of restricting the client site machines.

Problem 3. Shuffle Issues

(a) As you are able to see the traffic going in and out of the mix, by counting packets in, you can figure out the threshold number for the size of the anonymity set that the mix sends at. With this, you can significantly reduce the size of the anonymity set by waiting for the target user to send a message in, then sending your own messages to the mix until the threshold number is reached. On the other side, you can fiiter out the messages you sent, as you already know the destination of them. Using this, it is possible to eventually reduce the anonymity set to just the target user, for which you would know their destination.

(b) If a user wants to anonymously send one email, the mix can raise the size of the anonymity set by adding a bunch of cover traffic. This would make the anonymity set larger and more anonymous, protecting the user. In addition, by injecting a certain amount of cover traffic every time, even if an adversary can launch the attack in part a.), the anonymity set will always be at least a certain size of traffic that the adversary cannot filter out by virtue of knowing its destination and source.

(c) No. The cover traffic would be largely random, and if the same user is talking to the same recipient multiple times, a probibalistic attack can be launched to spot this pattern in the random traffic.