Software Puzzle: A Countermeasure to Resource-Inflated Denial-of-Service Attacks

Denial-of-service (DoS) and distributed DoS (DDoS) are among the major threats to cyber-security, and client puzzle, which demands a client to perform computationally expensive operations before being granted services from a server, is a well-known countermeasure to them. However, an attacker can inflate its capability of DoS/DDoS attacks with fast puzzlesolving software and/or built-in graphics processing unit (GPU) hardware to significantly weaken the effectiveness of client puzzles. In this paper, we study how to prevent DoS/DDoS attackers from inflating their puzzle-solving capabilities. To this end, we introduce a new client puzzle referred to as software puzzle. Unlike the existing client puzzle schemes, which publish their puzzle algorithms in advance, a puzzle algorithm in the present software puzzle scheme is randomly generated only after a client request is received at the server side and the algorithm is generated such that: 1) an attacker is unable to prepare an implementation to solve the puzzle in advance and 2) the attacker needs considerable effort in translating a central processing unit puzzle software to its functionally equivalent GPU version such that the translation cannot be done in real time. Moreover, we show how to implement software puzzle in the generic server-browser model.

EXISTING SYSTEM:

- DoS and DDoS are effective if attackers spend much less resources than the victim server or are much more powerful than normal users. In the example above, the attacker spends negligible effort in producing a request, but the server has to spend much more computational effort in HTTPS handshake (e.g., for RSA decryption). In this case, conventional crypto-graphic tools do not enhance the availability of the services; in fact, they may degrade service quality due to expensive cryptographic operations.
- The seriousness of the DoS/DDoS problem and their increased frequency has led to the advent of numerous defense mechanisms.
- As the present browsers such as Microsoft Internet Explorer and Firefox do not explicitly support client puzzle schemes, Kaiser and Feng developed a web-based client puzzle scheme which focuses on transparency and backwards compatibility for incremental deployment. The scheme dynamically embeds client-specific challenges in webpages, transparently delivers server challenges and client responses.

DISADVANTAGES OF EXISTING SYSTEM:

- Puzzle is designed based on client's GPU capability, the GPU-inflation DoS does not work at all. However, we do not recommend to do so because it is troublesome for massive deployment due to (1) not all the clients have GPU-enabled devices; and (2) an extra real-time environment shall be installed in order to run GPU kernel.
- However, this scheme is vulnerable to DoS attackers who can implement the puzzle function in real-time.
- Existing systems are not dynamic.

PROPOSED SYSTEM:

- In this paper, software puzzle scheme is proposed for defeating GPU-inflated DoS attack. It adopts software protection technologies to ensure challenge data confidentiality and code security for an appropriate time period, e.g., 1-2 seconds. Hence, it has different security requirement from the conventional cipher which demands long-term confidentiality only, and code protection which focuses on long-term robustness against reverse-engineering only.
- Since the software puzzle may be built upon a data puzzle, it can be integrated with any existing server-side data puzzle scheme, and easily deployed as the present client puzzle schemes do. Although this paper focuses on GPU-inflation attack, its idea can be extended to thwart DoS attackers which exploit other inflation resources such as Cloud Computing.
- By exploiting the architectural difference between CPU and GPU, this paper presents a new type of client puzzle, called software puzzle, to defend against GPU-inflated DoS and DDoS attacks.

ADVANTAGES OF PROPOSED SYSTEM:

- SSL/TLS protocol is the most popular on-line transaction protocol, and an SSL/TLS server performs an expensive RSA decryption operation for each client connection request, thus it is vulnerable to DoS attack.
- Our objective is to protect SSL/TLS server with software puzzle against computational DoS attacks, particularly GPU-inflated DoS attack. As a complete SSL/TLS protocol includes many rounds, we use RSA decryption step to evaluate the defense effectiveness in terms of the server's time cost for simplicity.
- The software puzzle scheme dynamically generates the puzzle function.

SYSTEM SPECIFICATION

Hardware Requirements:

- System : Pentium IV 3.5 GHz.
- Hard Disk : 40 GB.
- Monitor : 14' Colour Monitor.
- Mouse : Optical Mouse.
- Ram : 1 GB.

Software Requirements:

- Operating system : Windows XP or Windows 7, Windows 8.
- Coding Language : Java – AWT,Swings,Networking
- Data Base : My Sql / MS Access.
- Documentation : MS Office
- IDE : Eclipse Galileo