Computers & Security (2005) **24**, 76–82

ELSEVIER

Computers & Security

www.elsevier.com/locate/cose

# Improvements on the WTLS protocol to avoid denial of service attacks

**Ruishan Zhang***, **Kefei Chen**

*Department of Computer Science and Engineering, Shanghai Jiaotong University, 1954 Hua Shan Road, Shanghai 200030, PR China*

Received 7 November 2003; revised 12 September 2004; accepted 7 October 2004
Available online 28 January 2005

**KEYWORDS**
WTLS;
WAP;
DoS attacks;
Client cookies;
Client puzzles

**Abstract** The current WTLS protocol is closely modeled after the well-studied SSL protocol. However, since some differences exist between these two protocols, even if the SSL protocol is secure, the WTLS protocol may not.

We propose three kinds of possible Denial of Service (DoS) attacks on the existing WTLS protocol, which can be categorized into two types: memory exhaustion attacks and CPU exhaustion attacks. The first and the second kinds of attacks belong to memory exhaustion attacks, and the third kind of attack is a CPU exhaustion attack.

Not only wireless network clients but also Internet clients can launch these three kinds of attacks, which are very simple and effective. Since Internet clients are more powerful in network bandwidth and CPU resources, damages made by these attackers are more serious, which can even make the WTLS server stop providing services for legitimate clients.

Client cookies, client puzzles and an application timer is used to improve the current protocol, and our improvements are secure against such attacks.


## Introduction

The primary goal of the WTLS protocol is to provide privacy, data integrity and authentication between two communicating applications. The current WTLS protocol (WAP Forum, 2001) is closely modeled after the well-studied Secure Socket Layer (SSL) protocol (Freier et al., 1996). However, some differences exist between these two protocols. The SSL protocol operates on the top of the reliable TCP protocol, whereas the WTLS protocol runs on the top of unreliable connectionless Wireless Datagram Protocol (WDP). So even if the SSL protocol is secure, the WTLS protocol may not.

In this paper, we propose three kinds of DoS attacks on the WTLS protocol. All of them are easy

* Corresponding author.
*E-mail addresses:* zhang-rs@cs.sjtu.edu.cn (R. Zhang), chen-kf@cs.sjtu.edu.cn (K. Chen).

0167-4048/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cose.2004.10.002

to perform. However, the damages are serious. For example, though based on the unreliable connectionless WDP protocol, the WTLS protocol only specifies the timeout timer and the retransmission timer for the client and doesn't specify any timer for the server, which makes the protocol susceptible to DoS attacks, and exhausts the server's memory. To launch such an attack, an attacker just needs to use a forged IP address and sends some messages. Even in such a simple way, the attacker can exhaust the server's memory and make it fail to provide services for legitimate clients.

DoS attacks on protocols and corresponding resistant methods in wired networks are not new topics. The SYN flooding attack against the TCP connection protocol on the Internet was described, e.g., in CERT Coordination Center (1996). The possible remedies were analyzed in detailed in Inc. Berkeley Software Design (1996) and Cox (1996). Some general techniques, such as the two-phase authentication, client puzzles and client cookies, to resist DoS attacks on protocols were presented and studied (Karn and Simpson, 1999; Aura et al., 2000). For example, Cookies have been preciously used in the Photuris protocol (Karn and Simpson, 1999) and in the IKE protocol. Client puzzles were used to resist DoS attack in Aura et al. (2000). Furthermore, some DoS-resistant protocols were designed (Karn and Simpson, 1999; Aiello et al., 2002). Meadows (1999) formalized the idea of gradually strengthening authentication. So far, the basic principles are well studied and become clearer (Karn and Simpson, 1999; Aura et al., 2000; Aiello et al., 2002). However, no one proposed the method of DoS attacks on the WTLS protocol until now.

So in this paper, we firstly present three possible kinds of DoS attacks on the current WTLS protocol. And then, client cookies and client puzzles mentioned above are applied to improve the current protocol.

The remainder of this paper is organized as follows. In the next section, we review the WTLS protocol. Then, three kinds of DoS attacks on the WTLS protocol are proposed which is followed by three improvements on the current WTLS protocol. Further, we analyze the improved protocol. Finally, a conclusion is given in last section.

## The WTLS protocol

The WTLS protocol is similar to the SSL protocol. When the WTLS client wants to establish a secure connection with the WTLS server, the client and the server need to exchange a sequence of messages, which is called a handshake. There are three types of handshakes: a full handshake, an optimized handshake and an abbreviated handshake. For simplicity, we just introduce the full handshake here.

There are five steps in a full handshake, as are depicted in Fig. 1. In the first step, the client sends a ClientHello message. In the second step, the server sends ServerHello, Certificate*, ServerKeyExchange*, CertificateRequest* and ServerHelloDone messages to the client. In the third step, the client sends Certificate*, ClientKeyExchange*, CertificateVerify*, [ChangeCipherSpec] and Finished messages. In the fourth step, the server sends [ChangeCipherSpec] and Finished messages. In this step, the handshake is complete and the secure connection is established. In the fifth step, the client and the server begin to exchange application layer data under the negotiated secure connection. In Fig. 1, "*" indicates optional messages that are not always sent, and "[]" means that the ChangeCipherSpec message is not a part of the handshake protocol.

## DoS attacks

Before we describe DoS attacks, we should first point out that not only a wireless WAP client but also an Internet client have the ability to be an

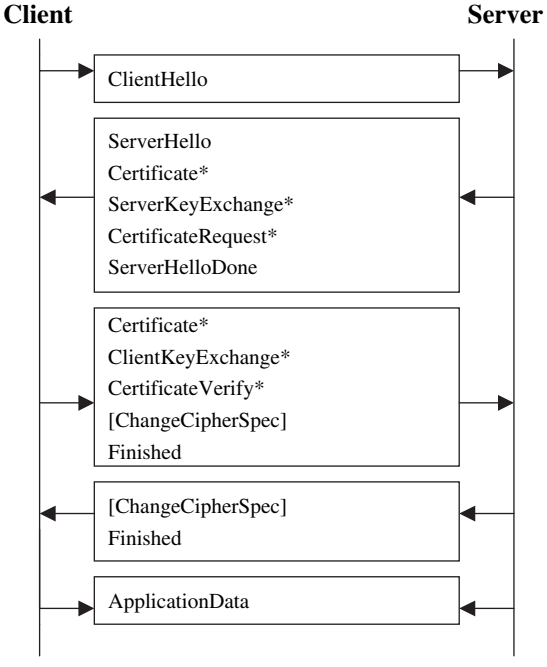


**Figure 1** A full handshake.