Speeding Up Caller ID Authentication

Mark Boutros

Computer Science Department, The College of New Jersey, Ewing,

New Jersey 08628,USA, boutrom1@tcnj.edu

Abstract. In recent years, the prevalence of Caller ID spoofing has resulted in many people losing all trust in the Caller ID system. In order to reintroduce trust and utility in the Caller ID system, an authentication mechanism should be introduced. Authentication mechanisms for Caller ID have been designed in the past, however many of these mechanisms do not provide the performance required to authenticate Caller ID efficiently enough to be consistently effective. The HTTP/3, QUIC, and TLS 1.3 protocols provide a significant performance improvement compared to prior versions of the HTTP protocol. Caller ID authentication can be made more efficient and effective by utilizing the novel UDP-based HTTP/3 versus the TCP-based HTTP/2.

Keywords: Caller ID, spoofing, authentication, HTTP/3, QUIC, TLS 1.3, UDP

1. Introduction

Caller ID (Caller Identification) is a functionality of telephony which informs the receiver who is calling. For numbers known to the receiver, Caller ID displays the contact name associated with the caller's phone number. For numbers unknown to the receiver, Caller ID displays the name of the area associated with the phone number's area code (the first three digits of the phone number).

There are a variety of ways to manipulate caller ID. Organizations with many conventional telephones may utilize private branch exchanges(PBX), devices which manipulate the caller ID so that each phone in the organization may have a unique caller ID. Individuals can manipulate their Caller ID when utilizing various Voice Over Internet Protocol (VOIP) applications. While PBX's are expensive and require significant upkeep, free VOIP applications are common and designed to be used by even inexperienced users. VOIP applications make it relatively easy for callers to change their Caller ID to virtually any number.

Unethically changing one's caller ID, or caller ID spoofing, has become increasingly common. Unethical callers use caller ID spoofing in order to convince the receiver to answer the phone when otherwise they would not. These callers deceive by posing as organizations such as insurance companies, warranty sellers, and government agencies. The goal of these deceptions is to obtain sensitive information such as Social Security Numbers(SSN's) and credit card information. Spoofing callers have two main tactics while utilizing caller ID spoofing. The more common tactic for unethical callers is to utilize numbers with the same area code as the receiver, making them believe the caller is nearby, even if this is not the case. It is extremely difficult for receivers to identify these kinds of spoofing calls before answering. The less common tactic for unethical callers is to utilize the exact number of the organization they are imitating. While it is easier for receivers to identify this second spoofing tactic, those receivers who do not identify it as spoof are more likely to fall prey to the scam and give up sensitive information.

The Federal Communications Commission(FCC) has made combatting unlawful robocalls and malicious caller ID spoofing a top consumer protection priority. The FCC and the Federal Trade Commission(FTC) both endeavor to enforce a number of laws pertaining to caller ID spoofing as well as certain automated telemarketing calls, unlawful acts which are often perpetrated together. The Truth in Caller ID Act (TICIDA) prohibits "knowing transmission of misleading or inaccurate Caller ID information 'with intent to defraud, cause harm, or wrongfully obtain anything of value'"[1]. The Telephone Consumer Protection Act (TCPA) prohibits "certain calls made using artificial or prerecorded voice to residential or wireless telephones as well as certain telemarketing calls". Additionally, the Do Not Call Implementation Act (DNCIA) authorizes the FTC to collect fees in enforcement of the Do Not Call registry, in consultation and collaboration with the FCC. From 2010 through 2018, the FCC collected nearly $250,000,000 in fees from violators of the TICIDA or the TCPA according to their 2019 *Report on Robocalls*[1]. While these collections will greatly help the FCC in further enforcement of these laws, the Commission admits that they account for a significant but small number of offenders and offenses. The report goes on to detail many of the complexities and challenges preventing spoofing and automated calls as well as potential solutions to these issues.

While the FCC and FTC are making progress in preventing spoofing calls, the situation currently remains virtually unchanged. The overarching changes to the telephony system proposed by the federal commissions will take time to be approved and then implemented. It therefore falls upon developers and individuals to combat spoofing calls. A 2017 paper called *A Mechanism to Authenticate Caller ID*[0] describes a proposed mechanism to authenticate caller IDs, thus giving individuals confirmation of a caller's identity in some cases and indication of spoofing calls in other cases. This user application-based, opt-in system has significant advantages compared to possible future actions taken by the FCC and FTC. This mechanism can be developed and implemented significantly faster than the network core approach taken by the federal agencies. More notably, this mechanism could be used anywhere in the world, while the FTC and FCC obviously can only operate in the United States. The proposed mechanism, called 'Trusted Caller ID', is quite promising, however more research and testing is needed in order to fully describe the authentication system as well as determine its feasibility and practicality.

In recent years, developments in internet protocol (IP) technology have greatly improved the potential performance and security of the Internet as a whole. HTTP/3, the third major version of the Hypertext Transfer Protocol is currently in the process of being implemented throughout the net. It has already been adopted by a number of major browsers, including Google Chrome and Mozilla Firefox. HTTP/3 is based on QUIC, a UDP based, transport layer protocol, which was originally developed by Google as Quick UDP Internet Connections in 2012. HTTP/3, QUIC, and Transport Layer Security (TLS) version 1.3, aim to provide significantly faster connection establishment, ubiquitous encryption, and improved congestion control.

This paper will show how HTTP/3, QUIC, and TLS 1.3 can significantly improve the performance and security of the authentication mechanism proposed in *A Mechanism to Authenticate Caller ID.* The rest of the paper is organized as follows. Section 2 outlines the challenges of Caller ID authentication in detail. Section 3 explains the operation of the authentication mechanism first outlined in the 2017 paper *A Mechanism to Authenticate Caller ID.* Section 4 describes HTTP/3, QUIC, and TLS 1.3 in greater detail as well as how they will improve the authentication process. Finally, Section 5 summarizes the findings of this paper and outlines future work to be done for this project.

2. Background (Challenges)

There are a number of factors that make authenticating caller ID uniquely difficult. The Public Switched Telephone Network(PSTN) is composed of a variety of devices and technologies. Originally, the PSTN was composed solely of analog phones utilizing the Plain Old Telephone Service(POTS), a circuit switched service. The POTS is a simple and archaic technology when compared to modern telephony. Although the simplicity of this system makes it difficult for malicious users to spoof their caller ID, it can be hypothetically done through Private Branch Exchanges(PBX). Caller ID spoofing is rare using PBX's because of the cost, however POTS being the basis of telephony leads to two main challenges in Caller ID authentication today[0]. First, the telephone system still contains many analog devices utilizing the POTS, meaning there is no inherent means of caller ID authentication in telephony. Second, since almost all users of PBX's are legitimate, any mechanism designed to authenticate caller ID must allow for legitimate caller ID manipulation via PBX's.

Newer technologies have been introduced to telephony in recent years, including cellular phones and Voice Over IP (VOIP). In cell phones, the caller ID is supplied by the Subscriber Identification Module (SIM), making it nearly as difficult to spoof caller ID as with POTS with PBX's. In both the cellular network and POTS, the caller ID is correlated with a physical object(ie. a SIM card). This makes it reasonable for the developers of telephony to allow for the caller to supply their own caller ID, as spoofing a caller ID in this way is costly and not scalable. The advent and proliferation of the internet changed this completely. In VOIP, a packet-switched technology, a caller can spoof their caller ID to display virtually any phone number to the receiver. Spoofing applications utilizing VOIP can automatically call an extremely large number of victims, each time utilizing a new Caller ID. Additionally, when using VOIP it is much easier to hide one's true identity from federal authorities by utilizing virtual private networks or other cryptography techniques.

Advancements in Caller ID spoofing technology and techniques have made it more prevalent than ever, while making it nearly impossible to detect. Though the FCC had collected nearly $250,000,000 in fees from 2010-2018 for Caller ID spoofing violations, this accounts for only 7 offenders[1]. The commission faces a number of organizational challenges that make it even more difficult for them to fight caller ID spoofing. First, as a US government organization, they can only prosecute offenders within the United States, allowing foreign spoofers to be 'untouchable' to the FCC. Second, the bureaucracy involved in the organization often gives offenders an opportunity to evade investigation long enough to 'outrun' the statute of limitations for offenses. The FCC itself admits, "Currently, the only certain way to determine whether a call is wanted or unwanted is to answer it or let it go to voicemail". These factors make it pertinent for private organizations and individuals to tackle the issue of caller ID spoofing to come up with a solution that is international, efficient, and scalable.

At this point, the sheer prevalence of caller ID spoofing makes it infeasible to systematically identify spoofing callers. Therefore, the only reliable way to reintroduce trust in the Caller ID system is to introduce a mechanism that enables callers to positively authenticate their identity. Authentication of Caller ID comes with its own unique set of challenges. First, the authentication mechanism must be built on an infrastructure separate from the PSTN, as this archaic system does not have the means for authentication and would take huge time and capital investment to upgrade. Second, the authentication mechanism must provide security such that the process is confidential and non-tamperable. Third, the authentication process must be extremely fast and efficient as to complete authentication before the call is answered while being scalable enough to serve a potentially large number of concurrent users. Finally, the authentication mechanism must be fully interoperable with all forms of telephony, including POTS(with or without PBX's), cellular networks, and VOIP.

3. Authentication Mechanism

In order to negate the issue of Caller ID spoofing, we introduce a mechanism to authenticate caller ID. This section will describe the general architecture of the mechanism first proposed *A Mechanism to Authenticate Caller ID*[0]. The purpose of this description is to give a general understanding of the mechanism to the reader, however it may serve as a blueprint for the development of protocol specifications or prototype applications.

The mechanism utilizes a third party to authenticate the caller's Caller ID for the receiver. At the application level, this results in three 'flows' of data [Fig. 1]. First, the caller authenticates themselves to the Third Party Authentication Server(TPAS) over the Internet. Second, the caller initiates the call to the receiver as normal. Third, the TPAS confirms that the caller's ID is authentic and communicates this information with the receiver. While the description of these three distinct flows is in order, they may not necessarily execute discreetly (there may be overlap in the execution of the flows). This mechanism is time-critical as the authentication must be completed before the ringing of the receiver's phone concludes, however ideally the authentication is completed significantly before this point.

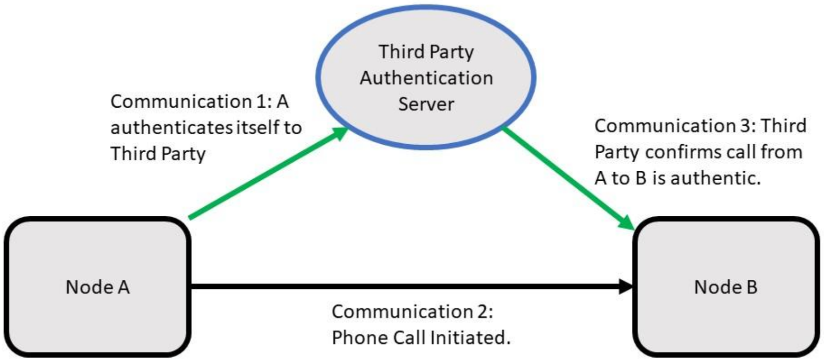

Third Party Authentication Server
Communication 1: A authenticates itself to Third Party
Communication 3: Third Party confirms call from A to B is authentic.
Node A
Node B
Communication 2: Phone Call Initiated.

Fig. 1. General Authentication Mechanism

The mechanism takes philosophical influences from a number of extant technologies. As an opt-in mechanism, both the caller and the receiver must complete a one-time onboarding process before use[Fig. 2]. During this process, the applicant sends identifying information such as phone number, IP address, and/or MAC address to the authority. The authority then verifies the information and generates a certificate, which is cryptographically tied to the applicant's device, the applicant's personal information, and the authority's device. This certificate is saved by the applicant to be used during the authentication process. This onboarding process closely resembles the SSL/TLS certification process taken by websites which utilize HTTPS.

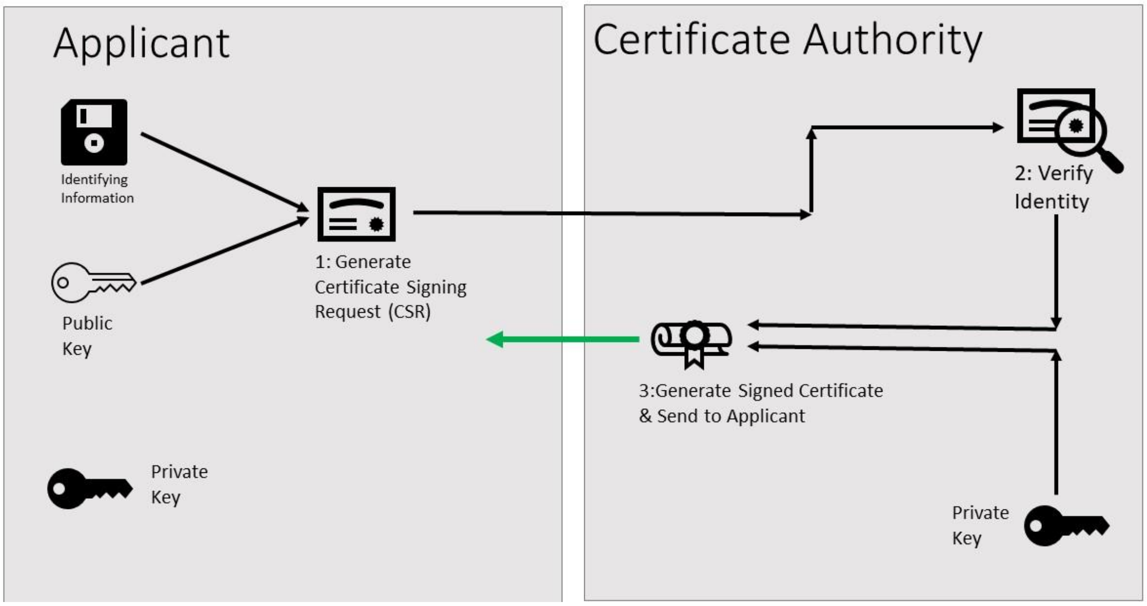


Fig. 2. HTTPS onboarding (certificate generation) process

The purpose of this mechanism is to prevent spoofing calls, however as stated, it is infeasible to reliably identify spoofing calls. By authenticating caller ID one can positively confirm the identity of a legitimate caller, however this mechanism does not confirm the presence of a spoofing caller. The social media site Twitter uses a system of authentication which inspires the output for this mechanism. Twitter's verification or 'blue checkmark' system is one of the most popular forms of authentication today. In this system, an account marked with a blue checkmark is known to be authentic. Alternatively, an account without the blue checkmark indicates it may be spoofed, however it may be authentic as well. The proposed authentication mechanism outputs states of authenticated/non-authenticated equivalent to the presence/absence of the blue checkmark in Twitter.

4. HTTP/3, QUIC, & TLS 1.3

HTTP/3 is the third major version of the Hypertext Transfer Protocol which is currently in the process of being standardized by the Institute of Electrical and Electronics Engineers(IEEE)[2]. HTTP/3 was designed for use with the novel Quick UDP Internet Connections (QUIC) protocol, a User Datagram Protocol (UDP) based transport protocol originally developed by Google in 2012. HTTP/3, QUIC, and Transport Layer Security (TLS) version 1.3, represent a major change in the structure of the Internet. The introduction of these technologies aims to significantly improve the overall performance and security of the Internet by bypassing the Transmission Control

Protocol(TCP) in favor of UDP [Fig. 3]. QUIC was designed with two major goals in mind[3]. The first goal was to add reliable delivery of data to the UDP while retaining its latency advantage over TCP. The second goal was to improve network performance by removing numerous inefficiencies caused by the nature of TCP. The new internet stack of HTTP/3 has allowed the developers of QUIC a number of advantages over previous stacks.

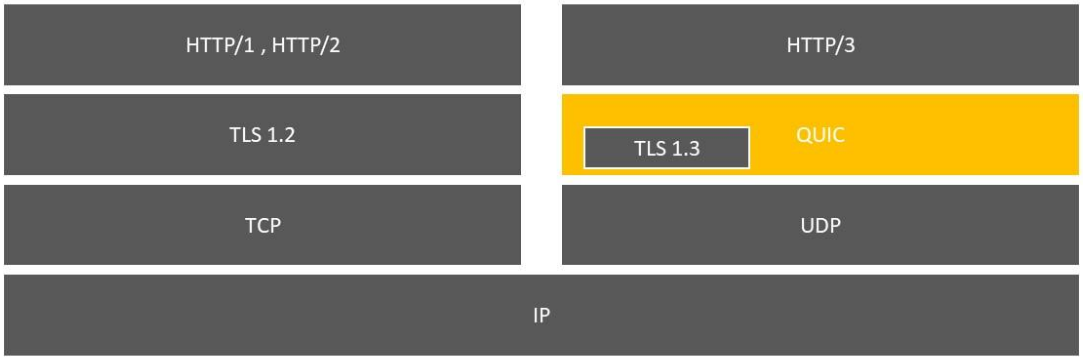
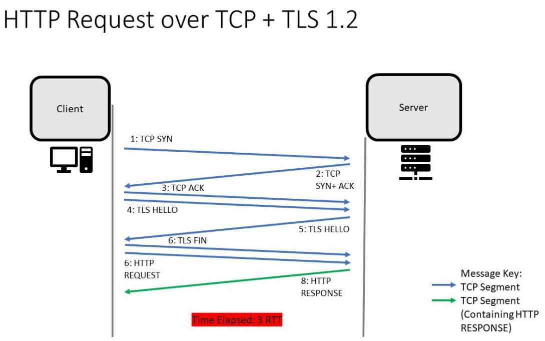


Fig. 3. Internet Stacks: HTTP/1 & HTTP/2 vs HTTP/3

QUIC implements reliable delivery of data at the transport layer, since UDP offers no guarantee of delivery. By implementing reliable delivery at the transport layer (as opposed to the network layer in TCP), QUIC moves a number of functionalities into the user space from the kernel space. Moving functionalities such as congestion control into the user space allows for much more development and nuance in these protocols.

QUIC utilizes a ‘secure-by-default’ philosophy, meaning that all communications over QUIC are encrypted. QUIC incorporates TLS 1.3, a hybrid cryptosystem which utilizes the Diffie-Hellman key exchange protocol[4]. TLS 1.3’s hybrid cryptosystem means that it utilizes the key exchange benefits of asymmetric encryption while also utilizing the performance benefits of symmetric encryption. One of the most significant benefits of QUIC over HTTP/2 is that the UDP-based protocol combines the cryptographic and transport handshakes in order to greatly reduce connection establishment time when compared to TCP connection establishment[Fig. 4a & 4b]. Additionally, QUIC and TLS 1.3 feature ‘0-RTT(Round Trip Time)’ connection resumption, which further reduces connection establishment time for certain connections [Fig. 4c]. ‘0-RTT’ connection resumption can be utilized whenever two hosts have saved a pre-shared key from a previous connection[7]. Applications can take advantage of this feature in certain situations when connections can be preempted. This increased efficiency in connection establishment results in 33% - 66% reduction in RTT and therefore total nodal delay in a simple HTTP request [Fig. 4].



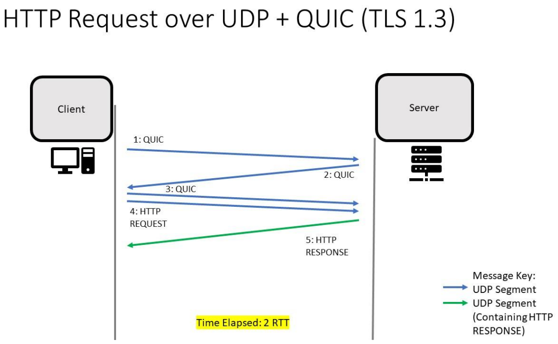


Fig. 4a. HTTP/2 Request(TCP). **Fig. 4b.** HTTP/3 Request(UDP).

Significant performance improvements can be achieved in Caller ID authentication by utilizing HTTP/3 & QUIC compared to HTTP/2. Analysing runtime differences between HTTP/3 and HTTP/2 is a challenging task due to the fact that QUIC has two modes of connection establishment.Those modes of connection establishment being default establishment and ‘0-RTT’ connection resumption.

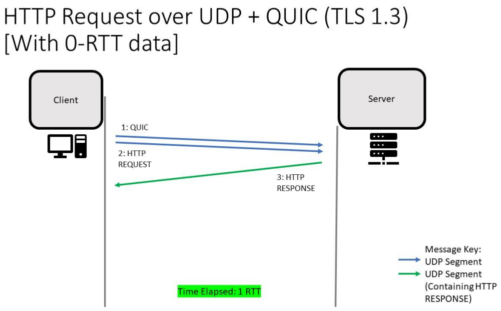


Fig. 4c. HTTP/3 Request(0-RTT).

The runtime of applications utilizing QUIC varies widely depending on which nodes have pre-shared keys and are therefore capable of ‘0-RTT’ connection resumption. Utilizing QUIC for the authentication mechanism described above results in a 66%-83% reduction in total RTT compared to TCP implementations. This reduction in RTT notably results in an equal reduction in total number of segments generated as well as total nodal delay. Additionally, the reduction in total nodal delay applies to propagation time, typically the most time-consuming aspect of nodal delay. While there still must be testing to determine the practical speedup of using QUIC to authenticate caller ID, the performance and security benefits of the protocol and its associated internet stack are clearly significant.

5. Conclusion

Caller ID spoofing has become extremely prevalent as a business model for many unethical callers due to the ease of perpetrating this crime combined with the limited ability of agencies to enforce Caller ID spoofing laws. This situation has come about in part due to the archaic nature of the telephony system, which still contains many inherent limitations. In order to reintroduce trust into the Caller ID system, a mechanism should be introduced which efficiently and effectively authenticates Caller ID. Utilizing the UDP-based QUIC protocol, in addition to TLS 1.3 and HTTP/3 significantly reduces connection establishment time compared to the earlier TCP-based protocols. Additionally, the standard security measures of QUIC from TLS 1.3 provide a high quality encryption mechanism. Implementing Caller ID authentication with HTTP/3 and QUIC would theoretically greatly improve the performance, efficiency, and effectiveness of authentication compared to HTTP/2, however further testing is required to determine the experimental performance gains.

Future work for this project should include experimental simulations in order to determine the actual performance benefits from utilizing QUIC for this authentication mechanism. Simulations would also help in discovering gains from aspects of QUIC not discussed in this paper, including connection multiplexing without head-of-line blocking, connection migration, and exportation of TLS 1.3 keying data[5]. Following experimental simulations, prototype and application development would be beneficial in furthering this project by assessing the challenges and benefits of QUIC in a real-life application setting. Finally, further research could investigate other time-sensitive network applications which would benefit from the inclusion of HTTP/3, QUIC, & TLS 1.3.

References:

0: ‘A Mechanism to Authenticate Caller ID’, Li, Faria, Chen, Liang

1: FCC Report on Robocalls:

https://www.fcc.gov/spoofed-robocalls

https://docs.fcc.gov/public/attachments/DOC-356196A1.pdf

2: Cloudflare on QUIC:

https://blog.cloudflare.com/http3-the-past-present-and-future/

https://blog.cloudflare.com/even-faster-connection-establishment-with-quic-0-rtt-resumption

https://blog.cloudflare.com/the-road-to-quic/

3: QUIC Internet Draft: https://tools.ietf.org/html/draft-ietf-quic-transport-34

4: TLS 1.3 RFC: https://tools.ietf.org/html/rfc8446

5: Keying Materials Exporters for TLS 1.3: https://tools.ietf.org/html/rfc5705

6: Augmented BNF for Syntax Specifications: https://datatracker.ietf.org/doc/html/rfc5234

7: Using Early Data in HTTP: https://www.rfc-editor.org/rfc/rfc8470.txt