The Undeniable Truth:
How Remote Attestation Circumvents Deniability Guarantees in Secure Messaging Protocols

Lachlan J. Gunn, Ricardo Vieitez Parra, N. Asokan, *Aalto University*

Deniability

During the 2016 US presidential election, attackers broke into John Podesta's e-mail account and published his mailbox via WikiLeaks; many messages could be authenticated by their DKIM signatures. After this, secure messaging apps saw a flood of new users: Signal, for example saw a 400% increase in downloads. One reason for this is that secure messaging applications, like Signal, promise cryptographic deniability: that when you send a message to someone, they can verify that it came from you but the protocol will not leave any trace that can be used to convince skeptical third parties who sent that message.

Cryptographic deniability has little value when the publisher of a message is trusted by the reader. This is the case in many real-world situations, for example:

- **Leaked documents published by a journalist.** In this case, the reputation of the journalist depends on the accuracy of their reporting, and so the reader may trust that the journalist has done due diligence, for example by comparing the information with that from other sources.
- **Plaintext messages presented in court by a forensic examiner.** In this case, it is the judge that must be convinced (or not) by the examiner, who obtains a plaintext by extracting it from the defendant's device. The judge can place a level of trust in the examiner, because the examiner gains relatively little from fabricating evidence, but has much to lose from being discovered doing so.

However, when the publisher is untrusted, deniability takes on greater importance. Little credence might be given to embarrassing documents when they are presented by a hostile propaganda organization, but presented alongside technical evidence of authenticity, they can be convincing, irrespective of how they are obtained. In an era where APT groups regularly obtain and publish email databases for political benefit, the importance of deniability cannot be overstated.

Remote Attestation

Many hardware platforms now include some form of *remote attestation*, by which the hardware proves some property of its state to another party. The attestation capabilities of many platforms permit one to prove the result of a computation. This has previously been used by the Town Crier protocol for the purpose of providing authenticated data feeds to smart contracts, but the far-reaching consequences for cryptographic deniability were not recognized.

Our attack is based on the principle that any locally-verifiable property can be attested to a third party. Thus, no protocol whose output can be attested can simultaneously provide both cryptographic deniability and sender-authentication of messages.

The Attack

The attack is implemented by executing the cryptographic state machine of an authenticated messaging protocol—in our demonstration, Signal—within an SGX enclave, a capability provided by Intel desktop processors since *Skylake*. The enclave produces a transcript of each session, including both the message contents and the origin of each message. It then produces an attestation over the transcript, proving to others that it is the product of a correct execution of an authenticated messaging protocol, and thus that the sender of each message did indeed send it.

Countermeasures

Several mitigations are available for this attack.

- **Online-deniable protocols.** These protocols are not vulnerable to our if the attacker's long-term identity key has been generated outside an attack TEE. This reduces the window of vulnerability to the period before the application has been installed and the identity keys generated. A remote attacker seeking to compromise a user's device and mount the attack against their contacts cannot do so without generating a new identity key and so alerting their victims.
- **Defensive remote-attestation.** Attestation attacks can be prevented by pre-emptively incorporating attestation into the messaging protocol itself. Before sending sensitive information, the sender demands an attestation from their interlocutor that demonstrates that their messages cannot be securely attested. This can happen in one of two ways: the recipient can use attestation to show that the key used for symmetric authentication *is not protected by a TEE*, and therefore that it can be used for forgery, or they can attest directly to the fact that they use a protocol implementation *that does not attest its output*. The former precludes protection of keys using a TEE, but its attested behavior is extremely simple and easily verifiable. The latter case allows the use of a TEE to protect protocol secrets, but requires verification of a large program, and is therefore more vulnerable to underhanded software development.
- **Avoiding sender-authentication.** Finally, we may design the protocol and surrounding system in such a way that the sender of a message is no longer fully machine-verifiable. One approach to this is to use a long-term identity key that differs for each other user. Any attestation of the protocol will only link messages to keys; because a user's identity key is different for each of their contacts, this prevents messages from being linked to a real person. In practice, this is difficult to implement: users expect that they can begin a conversation by looking up a user's identity, and unless carefully implemented, this may allow identity keys to be linked to their owners by combining the attestation attack with another similar attack against the key directory.

We recommend the use of online-deniable protocols as a countermeasure to this attack; they can substantially reduce the window of vulnerability to the period before the messaging application is installed, without introducing hardware dependencies, and without modification to the infrastructure surrounding the protocol. The next version of the Off-The-Record protocol, OTRv4, will be online-deniable and is under development at https://github.com/otrv4/otrv4.

Conclusions

Hardware supporting remote attestation is widely available and can be used to attack the deniability of any messaging protocol that provides sender-authentication. As a countermeasure, we recommend the use of protocols such as the upcoming OTRv4.

Further details are available in our technical report at https://ia.cr/2018/424.