Creating a Secure Connection Prior to Remote Attestation

Anna Fritz

November 17, 2021

Contents

Abstract

To promote cyber security, we must ensure networked communicating peers can be trusted before they share information. One way to establish this trust is through remote attestation where one party requests a communicating peer preform one or more measurements describing the state of their system. The target system subsequently preforms the measurements to generate evidence which is returned to the requester who evaluates the evidence to produce a trust decision. However, before any measurement is sent, the communicating peers must establish a secure networking connection to protect their traffic. This secure connection can be created using the Internet Security Association and Key Management Protocol (ISAKMP) which is a flexible framework that concludes with establishment of a security association. The security association is not only suitable for authentication but also serves as a framework to define applicable parameters for attestation such as the situation and lifetime. These parameters, and others, are critical to the success of the remote attestation routine. The following report describes the process and benefits of situating the ISAKMP routine prior to the remote attestation routine.

1 Introduction

Many internet users may want to establish trust in a communicating peer before sharing sensitive information like usernames and passwords to ensure their information is not shared with malicious entities. To motivate this thought, consider a banking example. When users log into their bank account online, they must be cognizant of the idea that their secret information may be available to malicious third-party entities. Simply put, they must question the trustworthiness of the network and their communicating peers to protect their personal information.

One way to solve this problem and establish trust in a remote peer, is through the process of remote attestation. During remote attestation, an appraiser uses measurements to verify a target is in a sound, trusted state. However, as the remote attestation framework stands, there is no mention of the need to establish a secure network connection prior to the attestation routine [**flexible**].

In works surrounding the attestation topic, there is some networking discussion. Specifically, in works like "TPM-based Network Device Remote Integrity Verification" the authors state that networking equipment must also be verified during the remote attestation procedure [**ietf-rats**]. In short, they state that evidence is needed from each piece of equipment in the networking configuration to ensure that all devices in the communication scheme should be trusted. Such devices include routers, switches, and firewalls. While it is necessary to include information about networking equipment in the trust decision, this work does not focus on establishing a secure connection but rather discusses networking equipment that must be proven secure.

Some authors do take into consideration the establishment of a secure connection prior to remote attestation [**linking˙remote**]. Yet, works like these describe a rudimentary secure connection where a target and a server exchange a known key through existing protocols like

SSL or IPSec. Using these protocols does not capture the situationally dependent parameters remote attestation needs to be successful.

With that, we show existing works on this topic are insufficiently detailed when describing the a mechanism to establish a secure connection between communicating peers prior to remote attestation. That is, previous works lack a mechanism to capture the situational demands of the attestation while also establishing a secure connection. Yet, it is the situational information that is essential to the success of attestation and therefore it must be uncovered.

Intended Contributions

To resolve the issue of establishing a secure connection prior to remote attestation, we introduce a modification to the attestation framework such that the existing networking standard, the Internet Security Association and Key Management Protocol (ISAKMP), should run before the remote attestation routine [**ISAKMP**, **prin˙remote**]. Using the ISAKMP framework is ideal because not only does it result in the establishment of a secure connection but also the framework maintains situational parameters that are necessary for the success of attestation. With that, our contribution is to combine two existing procedures to create a unique solution to the trust problem as currently such ideas about attestation and ISAKMP are designed to run independently.

In the following report, we present our plan to combine these topics to establish trust. We begin by presenting the existing background information in Section 2. In Section 3, we describe how the fundamentals of ISAKMP can be implemented to create a secure connection prior to remote attestation. This section also realizes why ISAKMP is the correct choice for this work. Finally, in Section 4, we discuss a motivating example about how the attestation version of ISAKMP is used in practice.

2 Background

With ISAKMP and remote attestation, gaining trust is a two step process. First, a secure networking channel must be established to protect subsequent messages. ISAKMP does this through the establishment of security associations. Second, the actual trust decision must be calculated and transmitted between the two parties. We choose to do this through remote attestation. Both of these preexisting ideas are explained further in the section below.

2.1 Remote Attestation

Remote attestation is a process by which a remote party can make a trust decision about a communicating peer [**prin˙remote**]. The party who would like to make the trust decision is the *appraiser* while the party who must prove their trustworthiness is the *target*. Remote attestation begins when the appraiser prompts the target to take a measurement [**copland**, **orchestrating˙layered**]. This prompt is called the request, as seen in Figure **??**. Once the target receives the measurement, they run it to generate evidence where the evidence describes the target's current, running state. This evidence is sent back to the appraiser who evaluates it to make a trust decision. Ultimately, the trust decision is presented as a binary value meaning either the target is trusted or not trusted. The result of the trust decision is agreed upon for some length of time, as the situationally determined.

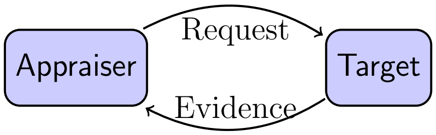


Figure 1: Remote attestation architecture showing an *appraiser* making an attestation request of a *target*.

In order to guide the development of the remote attestation architecture and its resulting systems, five guiding principles are established, as enumerated below [**prin˙remote**]. These principles are important to discuss as they motivate all remote attestation work.

1.Fresh Information
2.Comprehensive Information
3.Constrained Disclosure
4.Semantic Explicitness
5.Trustworthy Mechanism

To understand how the principles can be realized, a definition of their meaning is necessary. First, the principle of fresh information states that the target must provide information about the current, running system. Secondly, the target must provide comprehensive information. This means the target must provide enough detail so that the appraiser can make an informed trust decision. Thirdly, the principle of constrained disclosure says that the communicating peers should be able to enforce their privacy standards. Specifically, the target should have some way of enforcing its privacy standards so as not to divulge sensitive information through the shared evidence. This idea can be implemented with a privacy policy, and is discussed in further in the following section [**flexible**]. The fourth principle is the principle of semantic explicitness and it states the semantics should be logically defined. Finally, the principle of a trustworthy mechanism means that the underlying mechanism to establish trust is, in itself, trusted.

Using these principles, we arrive at the current attestation framework. However, this framework does not allow for the target or appraiser to enforce their privacy requirements. As it stands, without any secure connection, the appraiser selects their desired measurement and the target must run it or it is automatically labeled untrustworthy. To combat this problem, and meet the peer's privacy needs, the idea of negotiation is introduced and explained in the following section.

2.2 Negotiation

In order to meet the goals of comprehensive information and constrained disclosure, a negotiation routine must take place prior to attestation [**fritz**]. The intent of negotiation is to allow the target and appraiser to agree on a measurement that simultaneously protects the target's privacy standards while ensuring the appraiser has enough information to make an informed trust decision. That is, the communicating peers must find a measurement that is comprehensive enough to satisfy the appraiser's need for comprehensive information and yet constrained enough to satisfy the target's need for constrained disclosure [**flexible**]. Negotiation, as seen in Figure **??**, allows the target and appraiser to meet these goals.

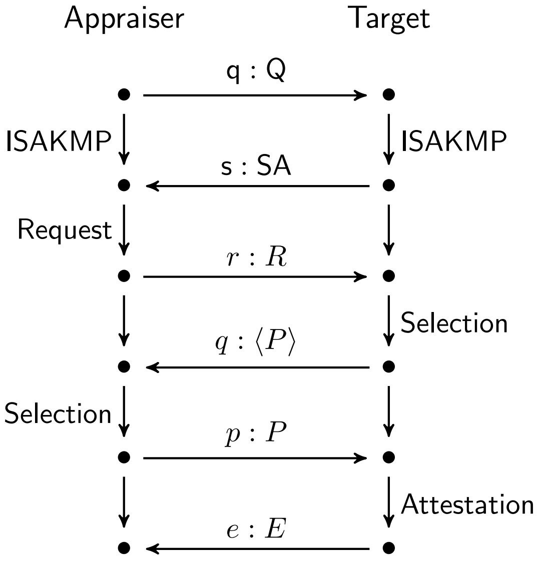


Figure 2: The negotiation procedure [**fritz**].

The negotiation routine begins with the establishment of a security association (SA) through the Internet Security Association and Key Management Protocol (ISAKMP) [**ISAKMP**]. That is, through ISAKMP, the situation, domain of interpretation, lifetime, and other situationally dependent criteria are established and recorded in a security association. Once that information is agreed upon, the appraiser sends the target a request. The request is one or more protocols the appraiser deems sufficient for attestation. It is situationally dependent and thus must take into consideration the information defined in the security association. Once the target receives the request, it applies its selection policy to generate a set of protocols that satisfy the request but do not expose sensitive information. This set, q, is called the proposal. Again, the information the target deems private is situationally dependent and thus must take into consideration details presented in the security association. Once generated, the proposal q is then sent back to the appraiser. Upon receiving the proposal, the appraiser can apply its selection policy to choose the best protocol for attestation. The resulting protocol, p, is then used in the attestation routine.

2.3 ISAKMP

One essential piece of ensuring the negotiation routine is successful is proper understanding of the Internet Security Association and Key Management Protocol (ISAKMP). That is, this networking standard allows two remote parties to obtain a secure communication link through the establishment of security associations (SA) [**ISAKMP**]. Each SA is unique to the communication context and can be conceptualized as a label of the agreed upon security services. Once instantiated, the SA protects subsequent traffic, thereby protecting all subsequent communications.

In general, the ISAKMP communication procedure occurs in two phases where the goal of the first phase is to establish protection for the second phase [**ISAKMP**]. To begin, the phase one procedure establishes the initial security association. This process is completed infrequently as it requires more networking resources and is therefore time and resource consuming [**ISAKMP**]. However, once the first phase is complete, the basic SA is established and valid for a predetermined amount of time. With that, negotiations for the second phase can begin. During phase two, the initial SA is used to exchange key material as well as other security protocols. Phase two takes place frequently and is not time consuming.

Often, the actual information being negotiated in phase two is implementation dependent. For example, an IPSec implementation that uses ISAKMP to establish a secure communication link requires predetermined implementation dependent fields to be negotiated [**rfc˙doi**]. Some of these fields include the authentication header (AH) transform, the encapsulating security payload (ESP) transform, and the key exchange protocol. In most implementations, fields like hashing algorithm, signing algorithm, and lifetime of the security association are negotiated.

To situate this work within the networking stack, we present Figure **??** below. It can be seen that ISAKMP should be implemented on top of the IP layer but below the domain of interpretation (DOI) [**thesis**]. In terms of an actual implementation of ISAKMP, the ISAKMP daemon is a user-level daemon that is registered with the operating system kernel [**thesis**]. Existing SA's are stored in the kernel's engine and can be used by the system through standard security mechanisms.

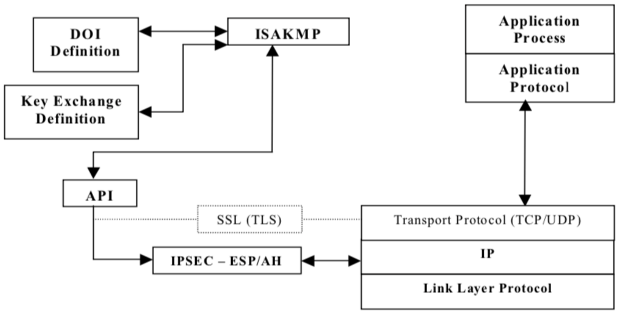


Figure 3: ISAKMP in network stack [**handshake**]

We believe it is important to note that ISAKMP does exist in practice today as part of the IPSec implementation [**securing**, **design˙IPsec**]. However, in IPSec, ISAKMP and IKE are combined to meet the networking goals of confidentiality, integrity, and authenticity [**IKE˙RFC**]. The details of this are beyond our scope but we note that even though the framework was developed in the late 1990's, it is still useful today.

This simple overview may not provide enough detail to grasp the fundamentals of ISAKMP so as to understand how it can be used prior to remote attestation to aid in meeting the attestation principles. Therefore, in following subsections, additional information is presented. This includes information about exchange types, payload types, the domain of interpretation (DOI), and a motivating example of protocol exchanges.

2.3.1 Exchange Types

Perhaps one of the most useful things about the ISAKMP framework is its flexibility. That is, to account for different communication scenarios, there are five types of exchanges: base exchange, identity protection exchange, authentication only exchange, aggressive exchange, and informational exchange [**ISAKMP**]. Most exchanges are responsible for swapping keys and authentication information yet others serve as a means to transport single messages. It is the order of messages, the amount of messages, and the contents of the messages, that makes each exchange type unique. The outline for each exchange is presented in Appendix Figures **??**, **??**, **??**, **??**, and **??** yet described in further detail below [**ISAKMP**].

First, the base exchange communication routine occurs with the transmission of 4 messages. During this exchange, keying information and authentication information are sent together. Coupling this information reduces the number of round trips but does not protect the communicating parties identities. This is because identification information is exchanged before a shared secret is established.

The identity protection exchange requires 6 messages total. The additional two messages allow keying information to be exchanged before the authentication information. This means this exchange can provide identity protection for the communicating parties as they can use the keying information as an established shared secret.

Next, the authentication only exchange operates with three messages where, as the name suggests, only the authentication information is exchanged. This incurs less of a computational burden, as there is no need to compute keys. Yet, with the lack of keys, none of the subsequent information can be encrypted and therefore it is not protected.

The fourth exchange type is the aggressive exchange where all security relevant information is exchanged in one message. This limits the number of round trips. Yet, the downside is that the initiator only sends one proposal and transform payload, meaning the responder has no chance to negotiate future parameters. However, unlike the authentication only exchange, this exchange does transmit keys to allow for protection of future messages.

Finally, ISAKMP has an informational exchange in which only one message is transmitted to allow communicating peers to send a notification or delete message. The notification message can be used for tasks like key recovery [**recovery**]. The delete message can be used to tell the responder to delete the security association. Either the party that initiated the connection or the party who responded may begin this type of exchange. Furthermore, the informational exchange may be sent at any time.

2.3.2 Payload Format

Each exchange is composed of a series of one or more payloads [**ISAKMP**]. To introduce a way to maintain state and allow for easier processing, each payload contains a fixed header. This header can be seen below in Figure **??**.

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td><td>19</td><td>20</td><td>21</td><td>22</td><td>23</td><td>24</td><td>25</td><td>26</td><td>27</td><td>28</td><td>29</td><td>30</td><td>31</td></tr>
<tr><td colspan="32">Initiator Cookie</td></tr>
<tr><td colspan="32">Responder Cookie</td></tr>
<tr><td colspan="8">Next Payload</td><td colspan="4">MjVer</td><td colspan="4">MnVer</td><td colspan="8">Exchange Type</td><td colspan="8">Flags</td></tr>
<tr><td colspan="32">Message ID</td></tr>
<tr><td colspan="32">Length</td></tr>
</table>

Figure 4: The ISAKMP payload header [**ISAKMP**].

For better understanding, the payload contents may require more description [**ISAKMP**]. First, the initiator and responder cookies are used for the entity that initiated the security association and the one who responded to it, respectively. These values are unique to each communication scenario. As it sounds, the next payload field is a numeric value that is used to identify the next expected payload. The possible values, and their meanings, can be seen in Figure **??**. Major and minor version are used to show compliance from the initiator and responder. They are automatically set to 1 for RFC 2408 (the ISAKMP communication routine) [**ISAKMP**]. Following the minor version, the exchange type field is used to describe what type of exchange is occurring. The values for this can be visualized in Figure **??**. Next, the flag bits provide flexibility for three options: encryption bit, commit bit, and authentication only bit. If the encryption bit is set to 1 then all headers are encrypted. If the commit bit is set to 1 then the party that did not set the bit must wait for the information only exchange. Finally, if the authentication only bit is set to 1 then the information from the commit bit's information only exchange is not encrypted. The message ID field is used to identify protocol state. The last header is the length field which is implemented to indicate the total messages in octets.

Payload Type	Value
Security Association (SA)	1
Proposal (P)	2
Transform (T)	3
Key Exchange (KE)	4
Identification (ID)	5
Certificate (CERT)	6
Certificate Request (CR)	7
Hash (HASH)	8
Signature (SIG)	9
Nonce (NONCE)	10
Notification (N)	11
Delete (D)	12
Vendor ID (VID)	13
Private Use	128-255

(a) Payload Type Values [**ISAKMP**]

Exchange Type	Value
NONE	0
Base	1
Identity Protection	2
Authentication Only	3
Aggressive	4
Informational	5
ISAKMP Future Use	6-31
DOI Specific Use	32-239
Private Use	240-255

(b) Exchange Type Values [**ISAKMP**]

Figure 5: Values for Payloads.

2.3.3 Domain of Interpretation

At its core, the goal of ISAKMP is to establish security associations (SA). One key component of the SA is the domain of interpretation (DOI). Within the DOI, communicating peers can group security related properties so that the peers can negotiate about their requirements [**rfc˙doi**]. In summary, the DOI is designed to:

1. define naming schemes
2. define the interpretation of the situation field
3. define useful security policies

In implementation, the negotiated domain of interpretation is located in the security association payload [**ISAKMP**]. To house such information, there is one security association payload per communication context. As seen in Figure **??**, the SA payload includes the DOI and the situation. The situation field describes the context of communication so that the peers can make informed policy decisions when negotiating during the establishment of the SA [**rfc˙doi**].

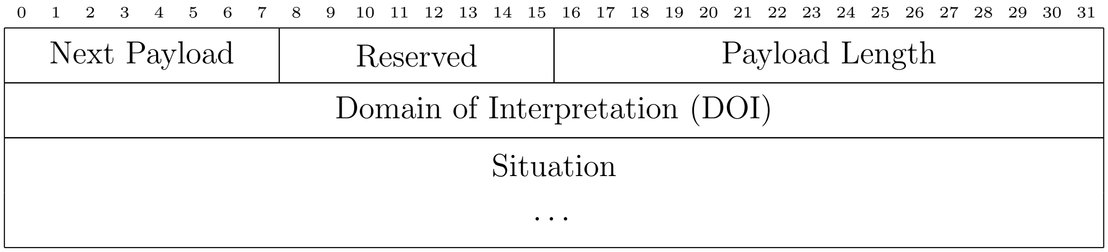


Figure 6: Security Association Payload. [**ISAKMP**]

One example of an implementation of the DOI occurs when using ISAKMP for IPSec. In this case, we observe the possible values for the situation to motivate its use. That is, the situation is realized as an octet bit mask with the following options: identity only, secrecy, and integrity [**rfc˙doi**]. For the identity only situation, the DOI can be recognized in an associated identification payload. For secrecy, the SA should be negotiated in an environment that requires secrecy. And, for integrity, the SA negotiation procedure should occur in an environment that requires integrity. This is simply one example that motivates the need and use of the situation field.

2.3.4 Protocol Exchange

To establish a security association, one or more payloads will be transmitted between the communicating peers [**ISAKMP**]. Specifically, there must be a single SA payload, which contains information describing the domain of interpretation and the situation as seen in Section 2.3.3. Following the SA payload, there maybe one or more proposal payloads. Each proposal payload is associated with a transform payload. The transform payload contains the specific security mechanisms to be used in the protocol.

To better understand this, consider the example in Figure **??**. As previously stated, the payload begins with the SA payload which contains the DOI and situation. The subsequent payloads contain two options for the resulting protection suite. The first option is protocol 1 with transform 1 and 2. This protocol is ESP and the transforms are 3DES and DES, respectively. The second proposed protocol (Prop1 Prot2) is the AH protocol with the transform SHA. With that, the responder chooses the resulting protection suite which will be either 3DES and SHA or DES and SHA.

<table>
<tr><th>0 1 2 3 4 5 6 7</th><th>8 9 10 11 12 13 14 15</th><th>16 17 18 19 20 21 22 23</th><th>24 25 26 27 28 29 30 31</th><th></th></tr>
<tr><td>NP = Nonce</td><td>Reserved</td><td colspan="2">Payload Length</td><td rowspan="3">SA
Pay</td></tr>
<tr><td colspan="4">Domain of Interpretation (DOI)</td></tr>
<tr><td colspan="4">Situation</td></tr>
<tr><td>NP = Proposal</td><td>Reserved</td><td colspan="2">Payload Length</td><td rowspan="3">Prop 1
Prot 1</td></tr>
<tr><td>Proposal # = 1</td><td>Protocol-Id</td><td>SPI Size</td><td># of Trans. = 2</td></tr>
<tr><td colspan="4">SPI (Variable)</td></tr>
<tr><td>NP = Transform</td><td>Reserved</td><td colspan="2">Payload Length</td><td rowspan="3">Tran 1</td></tr>
<tr><td>Transform # = 1</td><td>Transform ID</td><td colspan="2">Reserved2</td></tr>
<tr><td colspan="4">SA Attributes (3DES)</td></tr>
<tr><td>NP = 0</td><td>Reserved</td><td colspan="2">Payload Length</td><td rowspan="3">Tran 2</td></tr>
<tr><td>Transform # = 2</td><td>Transform ID</td><td colspan="2">Reserved2</td></tr>
<tr><td colspan="4">SA Attributes (DES)</td></tr>
<tr><td>NP = 0</td><td>Reserved</td><td colspan="2">Payload Length</td><td rowspan="3">Prop 1
Prot 2</td></tr>
<tr><td>Proposal # = 1</td><td>Protocol-Id</td><td>SPI Size</td><td># of Trans. = 1</td></tr>
<tr><td colspan="4">SPI (Variable)</td></tr>
<tr><td>NP = 0</td><td>Reserved</td><td colspan="2">Payload Length</td><td rowspan="3">Tran 1</td></tr>
<tr><td>Transform # = 1</td><td>Transform ID</td><td colspan="2">Reserved2</td></tr>
<tr><td colspan="4">SA Attributes (SHA)</td></tr>
</table>

Figure 7: Security Association establishment example [**ISAKMP**].

Once the responder chooses the resulting protection suite, the security association is established and used for subsequent communications. Showcasing this simple example highlights the flexibility of ISAKMP as many different proposals can be made to account for a variety of situations.

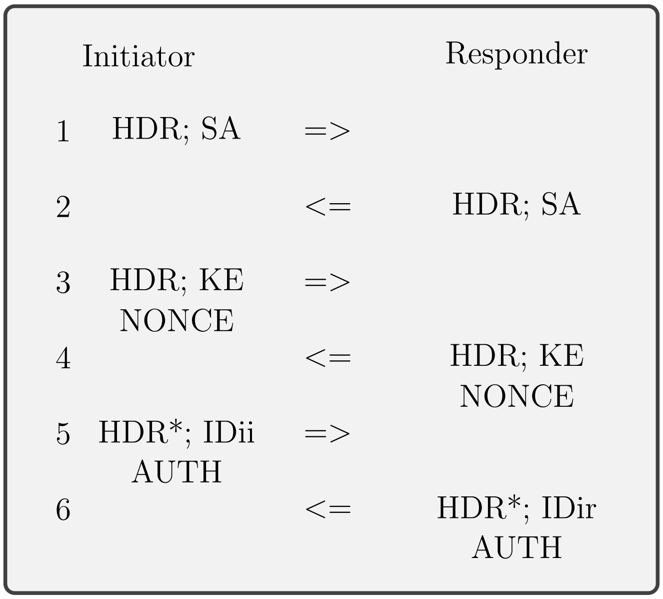
2.4 Motivation for this work

Currently, the remote attestation framework neglects to describe the mechanism for establishing a secure connection prior to the attestation routine. Authors who typically write about the attestation work tend to assume the connection away, stating that it already exists and further communication is taking place under the pretense of the secure agreement [**ietf-reference-interaction-models-04**]. In this work, we present a mechanism for creating a secure connection while describing how it can be implemented to capture the situational needs of the attestation framework.

3 The Use of ISAKMP for Secure Connections

There is little existing work that describes the networking requirements to establish a secure connection prior to remote attestation [**ietf-reference-interaction-models-04**]. As previously stated, we choose to implement ISAKMP to meet this need. Therefore, in this section, we discuss a practical ISAKMP exchange type for remote attestation as well as define accompanying ISAKMP fields to provide necessary contextual information for the success of the attestation routine.

When implementing ISAKMP, first we must to choose an exchange type. Perhaps the best fit for attestation is the Identity Protection Exchange so as to ensure that the identity of the target and appraiser are not vulnerable to third party attackers lingering on the network. Because we would like to mitigate attacks, the Identity Protection Exchange type is most effective, as the transmitted keys can be used to protect subsequent traffic. This can be observed below in Figure **??**.



	Initiator		Responder
1	HDR; SA	=>	
2		<=	HDR; SA
3	HDR; KE NONCE	=>	
4		<=	HDR; KE NONCE
5	HDR*; IDii AUTH	=>	
6		<=	HDR*; IDir AUTH

Figure 8: Identity Protection Exchange. Figure adopted from RFC 2408[**ISAKMP**].

It is important to understand that ISAKMP is not only useful for setting up a secure communication link but also for realizing situationally dependent information that impacts the negotiation routine. That is, the goal of negotiation is to obtain a sufficient measurement for attestation. This measurement is often situationally dependent meaning there must be some way to determine the situational needs prior to negotiation. To do this, the flexible nature of ISAKMP can be exploited to define valuable information.

With each exchange, there are a number of mutable payloads. This means parameters that are specifically useful for negotiation can be included by using the predetermined, provided payloads, like the ones seen in Figure **??** or using the private use payloads. In either case, ensuring the situationally dependent information is exchanged is necessary for the success of remote attestation. With that, we realize the following information must be determined through ISAKMP, as listed below:

- Identities
- Situation
- Hashing algorithm
- Signing algorithm
- Lifetime

First, identities must be determined prior to remote attestation. The peers must be aware of who they are engaging with because both the target and appraiser must uphold the goal of constrained disclosure. By providing this awareness through the identity field, the peers can then determine what policies are necessary to protect their information. The identities are also necessary to establish the situation, as discussed next. With any ISAKMP exchange, the identities are shared in the identification payload. Therefore, the ISAKMP framework already accounts for the transfer of identification information.

Following the identification information, the situation is needed to describe the context of communication as the negotiation procedure cannot be successful without such information. This contextual information can be realized with a label in the situation field. To motivate this, consider one such situation where attestation must be performed quickly. Because of this label in the situation field, negotiation may produce a simple measurement that the target can use to swiftly generate evidence. On the other hand, if a more robust measurement is required, the situation label could reflect that need. While the situation field does exist in ISAKMP, the values are implementation dependent and defined in the coming sections.

Next, both the hashing and signing algorithms must be agreed upon by the target and appraiser as they are used in the attestation process. Both of these have payload fields associated with them in the ISAKMP framework. Therefore, it will be easy to implement these features.

Finally, the lifetime component is traditionally used to communicate the duration the SA is valid. However, here, the lifetime is implemented in a separate field to convey how long the results of remote attestation procedure should be valid. In other words, the appraiser must state the duration of the trust decision as once expired, it is no longer valid. By having this field within the ISAKMP communication routine, we can meet the remote attestation goal of fresh information as once this value times out the information is no longer fresh. For implementation, ISAKMP is equipped with a lifetime payload which can be copied for our purposes.

Each one of these fields has a set of values which they can employ, as seen in Figure **??**. To start, the identity can either be T or A to represent the target or the appraiser. There may be additional identification information needed but that is abstracted to these values for now. Next, the situation can be one of three values. In most cases, the situation is normal, meaning attestation should proceed as designed. However, the situation may be dangerous meaning there is no room for negotiation of security parameters or other timely considerations. The connection must be established swiftly so that the trust decision can be made as quick as possible. The third option for situational values is the extended value. In this case, the appraiser wants a more comprehensive view of the target so the situation reflects the desire for a more detailed measurement. Following the situational values, there are fields for the hashing and signing algorithms. For hashing algorithms, these values may be one like MD5, SHA-256, or others. For signing algorithms, these values may be RSA, DSA, or any others. These values reflect current, acceptable algorithms. The last defined field is the lifetime component which depicts the length the trust decision is valid. It may be until the connection ends, every 5 minutes, every 5 hours, or some eluded amount of time.

Field	**Possible Values**
Identities	$\{A, T\}$
Situation	{Extended, Normal, Dangerous}
Hashing algorithm	{SHA-256, MD5 ... }
Signing algorithm	{RSA, DSA ... }
Lifetime	$\{ \text{t} \mid t > 0 \} \cup \{\text{terminating connection}\}$

Figure 9: Possible values for each field in the security association.

Because of the flexible nature of ISAKMP, we can easily fit these values into payloads to contextualize the communication. This is useful to meet the principles of remote attestation and also the negotiation requirements.

4 Examples

As previously discussed, ISAKMP is used to establish a security association prior to negotiation and remote attestation. We determined that the identities, situation, hashing algorithm, signing algorithm, and lifetime need to be negotiated and presented in the security association. To motivate the choice of these particular fields, we present some provoking examples. These examples fall into three categories representing three different situations where remote attestation may be useful. That is, a one-on-one attestation, a one-to-many attestation, and a many-to-one attestation.

For the first situation, consider the case of attesting to the state of the bank's software as a one-to-one example. In this case, the situation is normal as the attestation routine is not hurried due to some threat; it can operate normally. The values for hashing and signing here, and in all following examples, are extraneous. An interesting feature, in this case, is the lifetime value as we must capture the fact that the trust decision should be valid until the connection terminates. Logically, this is because when you establish connection to your bank's server, you want to be able to do tasks without being logged out. But, once you do log out, you need to be reassessed for trust. The values for all parameters can be observed in the table below.

Identities	T , A
Lifetime	Terminating connection
Hashing algorithm	MD5
Signing algorithm	RSA
Situation	Normal

For the second situation, consider the case where you, as a user from your home network, would like to a join a cooperate network. This one-to-many case requires you to attest to the state of the many cooperate servers. You may be interfacing with one server but that may just be one of many on the network. In order to provide your system with a complete understanding of the cooperate network's state, the attestation protocol is meticulous and therefore the situation is extended. Again, the lifetime of the trust decision expires upon terminating the communication as once you log off the cooperate network, the trust decision is no longer valid. The values for all parameters can be seen in the table below.

Identities	T, A
Lifetime	Terminating connection
Hashing algorithm	MD5
Signing algorithm	RSA
Situation	Extended

Thirdly, consider the same situation as example two but in reverse. This time, the cooperate network would like to attest to your state before you join the infrastructure. Instead of having an extended attestation situation, since the target is only one communicating peer, it is normal again. There is nothing dangerous or extended about the session, it is just many servers trying to asses the state of one device to make a trust decision. The values for the various fields can be visualized in the table below.

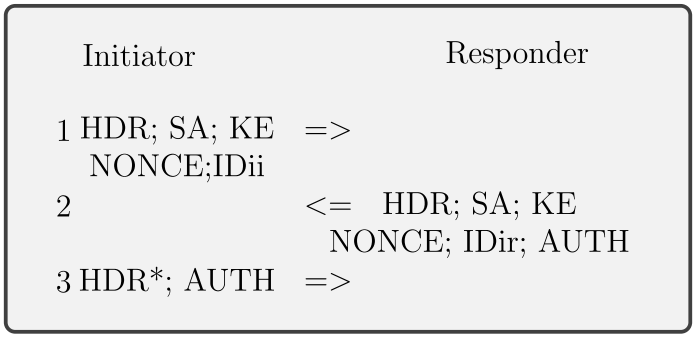
Identities	T, A
Lifetime	Terminating connection
Hashing algorithm	MD5
Signing algorithm	RSA
Situation	Normal

Hopefully these examples help to motivate the need for ISAKMP prior to negotiation and remote attestation. That is, these additionally proposed fields already exist within the ISAKMP framework as various payloads. With the flexible nature of the ISAKMP framework, we can morph them to fit the needs for negotiation and remote attestation to produce useful contextual information.

5 Conclusion

In this report, we have shown why ISAKMP may be an advantageous addition to the remote attestation routine. At first glance, ISAKMP can be applied to institute a secure connection prior to remote attestation through the establishment of security associations. Yet, its flexible nature also allows the framework to establish situationally dependent details. Taking this into consideration, we introduce the required identity, lifetime, hashing algorithm, signing algorithm, and situation payloads to the ISAKMP routine to situate the communication context. This allows the target to meet their goals of constrained disclosure as now, because of the identity and situation fields, they have an understanding of what information is considered private. Similarly, because of the contextual details the appraiser can speculate about what it means to have comprehensive information. Thus, by introducing this idea of implementing ISAKMP prior to remote attestation we not only set up a secure networking connection but also help meet the principles of attestation.

6 Appendix



```
  Initiator                    Responder

1 HDR; SA; KE     =>
  NONCE;IDii
2                 <=   HDR; SA; KE
                    NONCE; IDir; AUTH
3 HDR*; AUTH      =>
```

Figure 10: Aggressive Exchange [**ISAKMP**].

	Initiator		Responder
1	HDR; SA; NONCE	=>	
2		<=	HDR; SA; NONCE IDir; AUTH
3	HDR; IDii; AUTH	=>	

Figure 11: Authentication Exchange [**ISAKMP**].

	Initiator		Responder
1	HDR; SA NONCE	=>	
2		<=	HDR; SA; NONCE
3	HDR; KE IDii; AUTH	=>	
4		<=	HDR; KE IDii; AUTH

Figure 12: Base Exchange [**ISAKMP**].

	Initiator		Responder
1	HDR; SA	=>	
2		<=	HDR; SA
3	HDR; KE NONCE	=>	
4		<=	HDR; KE NONCE
5	HDR*; IDii AUTH	=>	
6		<=	HDR*; IDir AUTH

Figure 13: Identity Protection Exchange [**ISAKMP**].

	Initiator		Responder
1	HDR*; N/D	=>	

Figure 14: Notification Exchange [**ISAKMP**].