IKEv1 and IKEv2: A Quantitative Analyses

H.Soussi, M.Hussain*, H.Afifi, D.Seret

***Abstract*—**Key management is a vital component in any modern security protocol. Due to scalability and practical implementation considerations automatic key management seems a natural choice in significantly large virtual private networks (VPNs). In this context IETF Internet Key Exchange (IKE) is the most promising protocol under permanent review. We have made a humble effort to pinpoint IKEv2 net gain over IKEv1 due to recent modifications in its original structure, along with a brief overview of salient improvements between the two versions. We have used US National Institute of Technology NIIST VPN simulator to get some comparisons of important performance metrics.

***Keywords*—**Quantitative Analyses, IKEv1, IKEv2, NIIST.

I. Introduction

IPSEC is a suite of protocol designed by IETF [11] to provide security for IPv4 and IPv6. The security services include confidentiality, data authentification and data integrity. IPSec has become standard by default of the most of the IP VPN technology in the world. IPSec has three sub protocols: Authentication Header (AH), Encapsulating Security Payload (ESP) and Internet Key Exchange Protocol (IKE). AH assures integrity protection, ESP provide encryption services and optional integrity protection while IKE allows communicating entities to derive session keys for secure communication via a series of messages exchange.

IKE exists in two versions IKEv1 [2] and IKEv2 [4]. Although IKEv1 is flexible and contains diverse option its complexity remains a major problem for a wide spread implementation that is why constant evolution of the first version was inevitable. IKEv2 tries to make understanding easier than IKEv1 thanks to the number of exchange messages and the renewed choices of the encryption and authentification algorithms.

Effects of modification have been proven from a theoretical and mathematical point of view. But what about the quantitative gains that we get from there ameliorations? Despite the utmost importance of a quantitative comparison of IKEv1 and IKEv2, it is surprising that we find almost nothing in literature on this subject. This reason has pushed us to perform performance analyses of some parameter of IKEv1 and IKEv2.

Rest of the paper is arranged as follows. Section 2 gives an overview of the selective related work followed by a short description of IKEv1 and IKEv2 in section 3. Section 4 describes the main differences of both versions. Reasons that have lead us to perform this work are given in Section 5, followed by a presentation of the NIIST [3] simulator and a description of the simulation scenario. In section 8 we show the performance metrics and the reasons of their choice. Finally in section 9 we give results of our test and the analyses related to this work followed by the conclusion and references.

II. Related Work

R.Canetti and H.Krawczyk [10] have performed first IKE analysis based on pure mathematics. Their analyses have not pointed out the inherited implementation complexity found by many prominent researchers latter [13]. A non-mathematical analysis of IKE was done by Zhou [6] who suggests changes in the protocol. He also suggests an addition of payloads in main phase exchanges. However, this can lead to higher complexity into a protocol that is already considered complex [1].

Meadow [7] has used Navy Research Labs (NRL) protocol analyzer that is an expert system to analyze security protocols written in Prolog. IKE has been analyzed for secrecy, authentication and perfect forward secrecy. The authors have detected some minor ambiguities in IKE specifications that could have been source of attacks. Viewing the resources and efforts spent, the conclusions drawn from the analyses are trivial and had little effect on IKE evolution towards IKEv2.

K.Okhee and D.Montgomery [9] examined the behavior of IKE in a large scale VPN through various tests performed by the network simulator NIIST. Although their design objective includes large scale VPNs, but we have found several routing problems for a VPN simulation of considerable size (>50 SGs with 30 hosts each).

III. IKE

IKE (v1 & v2) is a set of protocols and mechanisms designed to perform two functions, creation of a protected environment (which includes peers authentication that are unknown to each other in advance) and to establish and manage Security Associations (SA) between the authenticated peers. IKE is heart of the IPSec because it not only controls the services to be offered to secure the traffic but also manages the whole range of different transform options available at different levels and at different granularity. IKE architecture is based on three other protocols, which are ISAKMP [RFC2408], OAKLEY [RFC 2412] and SKEME. It operates in two phases namely **phase1 and phase2**. Principal bricks of IKE are given below.

A. Security Association (SA)

Creation and management of a security association are the most fundamental concepts of the working of IKE and even IPSec. IKE can be considered as creator and manager of the SAs, while IPSec user of SAs. So big question arises, what a

H.Soussi is with the Ecole Nationale des Sciences Informatiques Tunis Tunisie.

M.Hussain and D.Seret are with the Université René Descartes UFR mathématique et informatique Paris-France.
{mhussain, seret, soussi}@math.-info.univ-paris5.fr

H.Afifi is with the Institute National de Télécommunication (INT) Evry France {afifi@int-evry.fr}.
*corresponding author

SA is? A SA is simply a contract between two entities to provide a minimum set of services. It can be bi-directional (as in phase1) or unidirectional (as in phase2). In case of unidirectional SA, which is often the case, we shall need two phase2 SAs to complete one communication. With the view point of a programmer a SA can be considered as a data structure containing the information on Security Policy Index (SPI), its state (alive or expired), authentication algorithm, sequence number and SA life time. Considering globally, an SA is a set of proposals. A proposal can be thought as a set of protocols and a protocol is, in turn, is a set of transforms. A transform is a set of algorithms.

B. Phase1-Main Mode

Here we have four messages (IKEv1 has 8 messages) exchanges between the initiator and the receiver. The purpose of this mode is to generate the shared secret from which other keys will be computed and authenticate the communicating peers.

C. Phase2-Quick Mode

The purpose of this phase to create an IPSec security association and to generate new keys. In IKEv2, this SA is denoted as Child-SA which is created as a result of Create-Child-SA request. This request may be launched by any of the party once phase one is completed. All messages in this phase are made secure due to the algorithms and keys negotiated in the first phase.

IV. Important Differences Of IKEv1 With IKEv2

All the changes which have been proposed and are being done in IKEv1 have, overall contributed very positively with respect to simplicity, flexibility, and security. Here are few distinguished points.

A. Flexibility and Simplicity

The first and most important difference between IKEv1 and IKEv2 is the lesser complexity and greater clarity for which IKE has always been blamed by many prominent researchers [1][13]. In IKEv2, number of phases and the number of messages to be exchanged during these phases has been significantly reduced.

B. Enhanced Security

The possibility of DoS type of attack was a major vulnerability of IKEv1 which has been removed by adding supplementary mechanisms. To make DoS attack harder, the responder may ask for a cookie to the initiator who has to assure the responder that this is a normal connection.

In fact large number of requests can be sent to a victim to waste its CPU and memory resources therefore a mechanism has been added in IKEv2 so that if a party suspects the incoming requests aren't genuine, it may verify the existence of correct, real ip addresses from which the request are being generated.

C. Re-structuring & New additions

A source of continued confusion in IKEv1 was the inter-relationship of attributes, transforms and proposals. Thanks to re-structuring of these terms in a hierarchical fashion that it is much easier to understand the whole story. In IKEv1, SA life times of communicating parties were negotiated in the beginning and both had to abide by this agreement. In IKEv2, there has been greater flexibility and each party is capable of choosing an SA lifetime of its choice independent of the other's.

V. Why Quantitative Analyses ?

Although there have been already theoretical and mathematical analyses their impact on the evolution of IKEv1 are minor. Due to the massive deployment, it is very important that the second version is not only analyzed theorically and mathematically but also it is compared quantitatively with the older one (IKEv1).

IKEv2 has a series of performance parameter, which have been either redefined or modified so it is important to demonstrate what is the net gain these modifications brought in. In this paper we have tried to present the relative impact of modifications with respect to IKEv1 on VPN's performance metrics and present our results in form of graphs. We will observe the behavior of the two versions with respect to these metrics. In these quantitative analyses we will focus mainly on the quantity of SAs created and the average delay taken to create those SAs, along with impact on bandwidth.

VI. NIIST

NIIST (NIST IPsec and IKE Simulation Tool) is an integrated Internet security simulation framework developed by National Institute of Standard and Technology. NIIST is implemented in Java and integrated in the Scalable Simulation Framework (SSF) and SSF Network Model (SSFNet) to provide an integrated Internet security-modeling framework in large-scale network. The SSF is a discrete, event-driven, scalable modeling framework and SSFNet is a collection of Internet modeling tools for simulating Internet protocols and networks. SSF places particular emphasis in scalability and high-performance for simulating very large networks.

The goal of NIIST is to enable relative performance characterizations of the impact of the variables above end-to-end applications. NIIST provides instrumentation to conduct detailed analyses of IPSec/IKE performance and its effect on end-to-end protocols such as TCP. NIIST is not designed for evaluating the underlying security properties of these protocol suites, and as such, abstracts actual cryptographic techniques away to only model their impact on performance.

VII. Simulation Scenario

In this simulation we consider following VPNs in which a security gateway connects a given site to every other site over IPSec tunnels. We use a network configuration of N sites, each consisting of M hosts and a single security gateway (SG). The hosts and SG at each site are connected by a 100Mbps link, the SGs are interconnected by 1,5 Mbps WAN links. A TCP client server application is used among the hosts. The configuration presented in this experiment provides N/2 client sites and N/2 server site. Each client site contains 50 host and each server site contains 10 host. Each 10 clients are connected to a server (Hosts and SGs are given in Data

Modeling Language (DML) syntax).

VIII. Performance Metrics

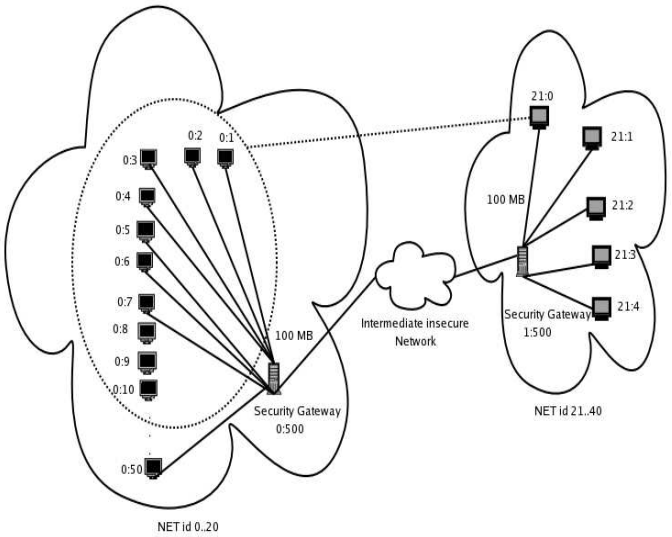

Fig. 1 Simulation Network Schema

NIIST is capable of producing performance measures at various protocol levels and observation points within a VPN. At the (SGs) we focus on the dynamics of IKEv1 and IKEv2 behavior. We are interested in the following performance metrics:

For each type of SA (Phase1/Phase2, initial/rekey), we examine SA establishment latency, which is a measure of time taken to establish the SA as seen by the initiator. IKE SA latency is measured as the time taken from sending the first IKE message and to receiving the last message. The metrics that have been chosen and that can affect the SA's creation are:

--VPN's dimension: the number of hosts that can communicate together through VPNs varies from one configuration to another. The number of SAs created, message and keys exchanged increase with VPN's dimension and as a result of this the overload on the network increases too. Due to this, it is important to see which of the two protocols react better to the growth of the host's number and the network overload.

--Bandwidth: effect of IKEv1 and IKEv2 on bandwidth consumption can be a major element for a quantitative analysis because the average of created SAs and the delay of creation of the SAs may be severely impacted by available bandwidth.

--Packet's size: Like the bandwidth, the size of packets exchanged through the connexion between the different SGs can effect the output of IKE, so it is important to take a look on the effect of the packet's size on the average and the delay of SA's creation to conclude on the advantages that bring IKEv2 compared to IKEv1.

Number of other metrics can be chosen to do quantitative analyses but in this study we have taken what we think are the most important to begin with.

IX. Results

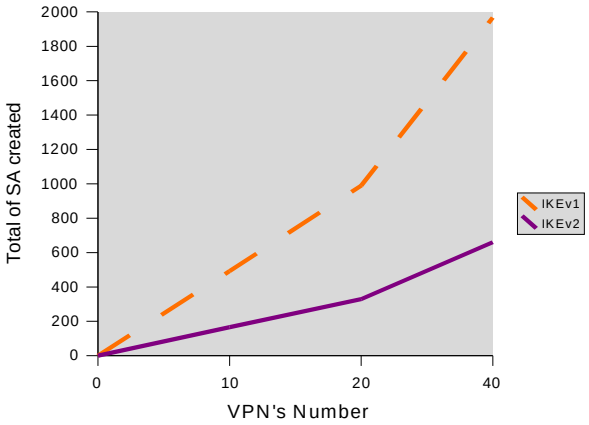

Fig. 2 Effect of IKEv1 and IKEv2 on VPN's number

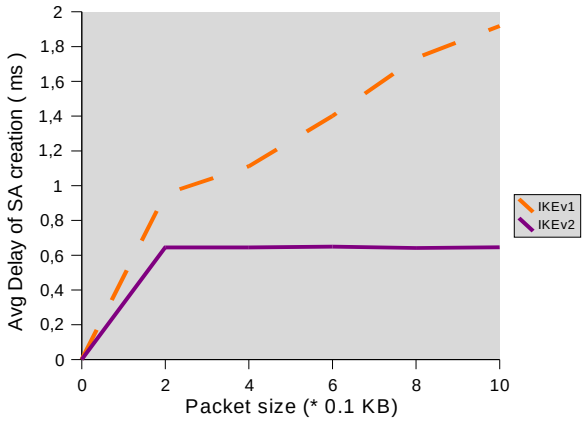

Fig. 3 Effect of IKEv1 and IKEv2 on packet size

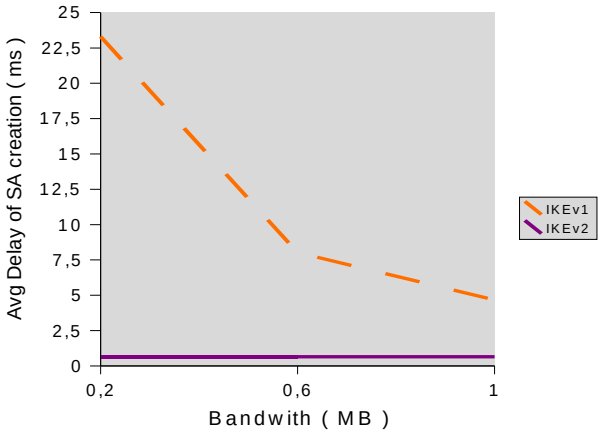

Fig. 4 Effect of IKEv1 and IKEv2 on Bandwidth

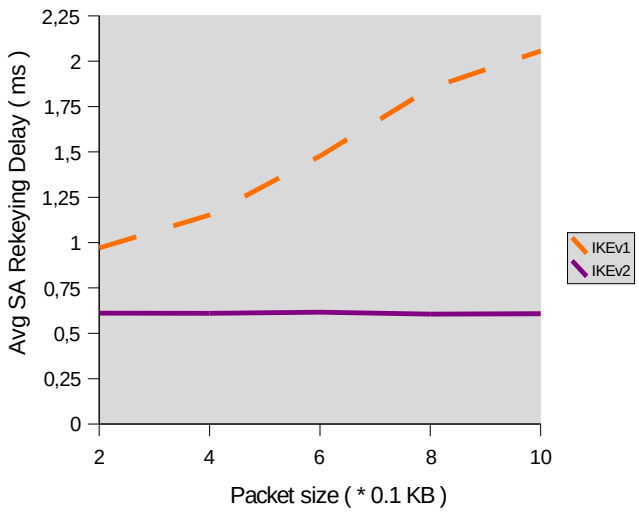

Fig. 5 Effect of packet size on IKEv1 and IKEv2 and Average Rekeying Delay

X. Analyses

In the first graph we can see the steep slop of IKEv1 comparing to IKEv2. This signifies that IKEv2 creates less SAs than IKEv1. Concretely, for 40 VPNs IKEv1 creates 1967 SAs and IKEv2 create 660 SAs. This demonstrate that the number of SA required to be created in IKEv2 is significantly less compared to IKEv1. This is due to the fact that IKEv2 provide less complexity and more flexibility reducing the traffic by overload compared to IKEv1. This is primarily due to the reduction of messages numbers and the reduction of the created SA's numbers in IKEv2.

As seen in the section IV the number of exchanged message have been decreased from IKEv1 to IKEv2 (10 message for IKEv1 and 6 message for IKEv2) so we can see in graph number 4 that the SA created in phase 1 takes more time then the one created in phase 2 and more the bandwidth increase and more the delay become constant. But for IKEv2 as the number of exchanged messages decreased in phase 1 and phase 2 the bandwidth doesn't totally affect the delay of the SA's creation. We can also see that the phase 1 require more SA's creation so the average delay is more important as we can see in the graph it take from 22 ms to 7,5 ms but the second phase begin we have lesser SA created so the delay lesser than for the first phase (7,5 ms).

We can see in the third graph that for IKEv1 the delay of SA creation increases as the packet size increase but IKEv2 keep the same average delay when the packet size increase. When the packet size increase the time for encrypting and treating packet header increase so when an SA creation is required the time taken to do this must increase. But the SA generated by IKEv2 are lesser than the one generated by IKEv1 so we can see that for any packet size the time taken to create those SA is always constant.

The rekeying SA offers a better security performance and reduces number of packets lost during transition but due to the redefinition of certain mechanisms (like ToS payload , choice of SA life time ,SPI uniqueness) of IKEv1 in IKEv2 less packet are lost and duplicated so there is less need to rekey Sas. That is why IKEv2 keep always the same SA creation delay as compared to IKEv1.

We can see that the rekeying delay of IKEv1 increases 1 ms to 2 ms showing the fact that after the establishment of the first phase 8 messages the second phase produces rekeyed SA and reproduce all the parameter of the previous SA so it takes more time for IKE v1 to rekey SAs than IKEv2 that have only 4 messages for the first phase.

XI. Conclusion

In this article we have used NIST VPN simulator (NIIST) to compare some performance parameters of two versions of IETF IKE protocol that is an essential part of IPSec suite. Our finding are that there are clear net gains in term of reducing network traffic, delay rekeying delay in phase 2 of IKEv2 as compared to IKEv1.We also have found that the modification brought to IKEv2 provided less complexity, traffic overload and more flexibility for large scale VPNs. As a next step we intend to identify more performance metrics and make a thorough comparison in real world scenario. Also we plan to develop a freeware VPN monitor based on IETF VPN MIB for a complete and comprehensive IP VPN monitoring for research community.

Acknowledgment

H.Soussi is grateful to Madame Dominique Seret for welcoming him in her laboratory and providing him the necessary equipment for doing his job in the best conditions. Also Monsieur Hussain and Monsieur Afifi deserve his sincere thanks for helping him during this work with their useful discussions.

References

[1] Ferguson, Niels, and Schneier, Bruce, “A Cryptographic Evaluation of IPSec”, http://www.counterpane. com, April 1999.
[2] D. Harkins and D. Carrel, The Internet Key Exchange (IKE) , RFC 2409, November 1998.
[3] http://www.antd.nist.gov/niist/
[4] C. Kaufman, Editor, Internet Key Exchange (IKEv2) Protocol , draft-ietf ipsec-ikev2-17.txt, September 23, 2004
[5] http://www.ssfnet.org/homePage.html
[6] J Zhou, Kent Ridge, ‘Further analysis of the Internet key exchange protocol’ Digital Labs, 21 Heng Mui Keng Terrace, Singapore, Computer Communications 23 (2000) 1606–1612
[7] Catherine Meadows, ‘Analysis of the Internet Key Exchange Protocol Using the NRLProtocol Analyzer 1999-Naval Research Laboratory Washington, DC 20375 Code 5543
[8] M. Hussain, I.Hajjeh, H. Afifi, D. Seret,”Tri-party IKEv2 in Home Networks”, ICACT 07 Seoul, South Korea.
[9] K.Okhee,D.Montgomery,”Behavioral and Performance Characteristics of IPsec/IKE in Large-Scale VPNs”,www.antd.nist.gov/niist
[10] Ran Canetti and Hugo Krawczyk ‘Security Analysis of IKE's Signature-based Key- Exchange Protocol ‘Crypto'03 (LNCS Series, Vol. 2729)].
[11] RFCs 2401, 2402, 2403, 2406, 2409, 2411
[12] Michael S; Borella, ‘Methods and protocols for secure key negotiation using IKE’ 3Com www.3com.com
[13] Perlman, R. and Kaufman, C. “Key Exchange in IPSec: Analysis of IKE”, IEEE Ineternet Computing, Nov/Dec 2000.