Distributed Monitoring of Self-Configuring Virtual Private Networks

Michael Rossberg Michael Grey Markus Trapp Franz Girlich Guenter Schaefer
Ilmenau University of Technology
{firstname.lastname}@tu-ilmenau.de

I. The Case for Self-Configuring VPNs

The cheap and globally available communication infrastructure of the Internet, makes it more and more interesting for companies and governments to also exchange private information over it. However, the often deployed manually configured Virtual Private Networks (VPNs) have certain limitations: While cryptography can ensure the confidentiality, integrity and authentication of transmitted data, the availability of globally accessible VPN is puzzling and will become an issue of increasing importance. This is due to the usual manual configuration approach, where a central VPN concentrator, a so-called *Hub*, is used to connect all participants (*spokes*). Centralized approaches like this hub-and-spoke architecture are very susceptible to attackers who control or rent botnets. Recently, powerful Denial-of-Service (DoS) attacks have even led to failures in the critical infrastructures of Estonia and the Republic of Georgia, and the threat of DoS-attacks had been reported to be used even in extortions.

In order to reduce this risk for VPNs, an entire paradigm shift in the management of such infrastructures is required. As distributed VPNs, which promise better resilience properties [1], cannot be reasonably configured by manual interaction, such VPNs must be automatically configured. Nonetheless, besides providing potentially better resilience against DoS-attacks, self-configuring VPNs may have also other advantages. They may be easier to deploy, allow for a more efficient integration of mobile users, and a more robust reaction to transport network failures. Furthermore, the number of security relevant configuration errors can be reduced, e.g., due to typos in addresses or subnet masks.

II. Secure OverLay for IPsec Discovery (SOLID)

One of such approaches is our **S**ecure **O**ver**L**ay for **IP**sec **D**iscovery (SOLID) system [2], which contains designated mechanisms for topology control, the discovery of VPN participants, and the routing of traffic. By embedding a Chord-like ring structure [3], [4] into arbitrary transport networks (see Fig. 1), it creates highly scalable VPNs with strong robustness properties. Within SOLID, a sample-based mechanism to select potential cross-connections allows to create structures with as few proactively established security associations as possible, in order to work efficiently, even if the tunnel setup requires multiple seconds due to the use of smart cards. One of the key-features is the usage of only locally available information in any of the VPN nodes, i.e., not even routing information needs to be broadcasted in the VPN in order to quickly adapt in the case of failures. Furthermore, data may be rerouted over other

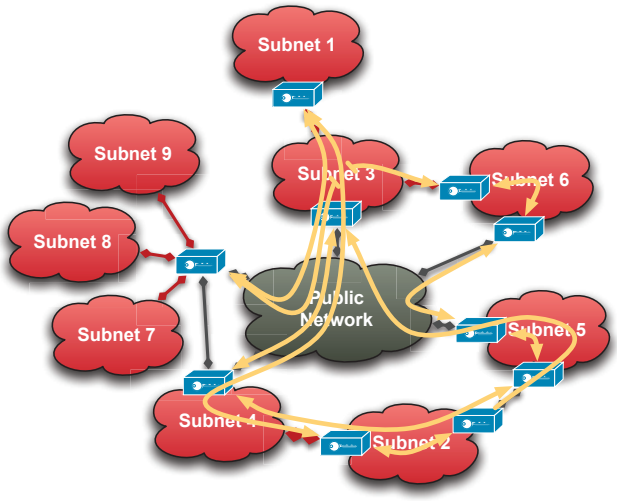

Fig. 1. Mapping of a virtual Chord-like VPN structure into a complex topology of the transport network

VPN participants if there is a partial failure in the transport network.

Nonetheless, to allow for administrators to detect and mitigate failures within the network, also such a self-configuring VPN requires some kind of monitoring. Furthermore, as the reactions of the introduced complex and distributed system must be also realizable by administrators, such a guidance is needed. However, in order be as secure as manually configured IPsec infrastructures, SOLID deploys a mandatory end-to-end IPsec protection for all IP traffic. Thus, the commonly deployed techniques, such as querying devices by Simple Network Management Protocol (SNMP), will perform bad as an administrative device would have to establish security associations to each queried device. Therefore, SOLID deploys a novel distributed monitoring approach that is based on a multi-tier architecture, as also depicted in Fig. 2. Herein,

- *VPN nodes* gather local statistics by observing their own state as well as passively monitoring their security associations. For example, by comparing the packet counters and replay counters of each security association the respective packet loss rates are estimated.
- *Aggregation Proxies* in different parts of the VPN then gather the available information securely. This is done by directly contacting some of the VPN nodes, which in turn automatically form a reverse multicast tree to also collect information from more distant nodes. In order to prevent

978-3-901882-50-0

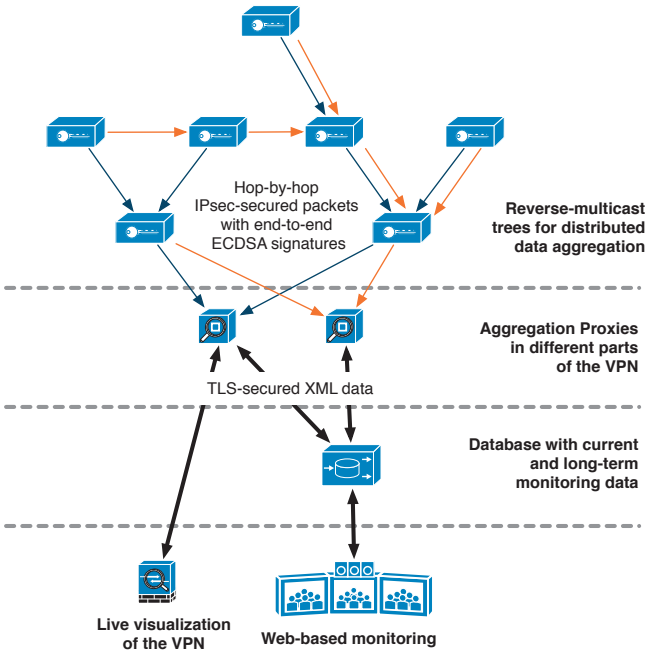

Fig. 2. Multi-tier architecture to securely retrieve VPN health information

this data to be accessible to illegitimate users, requests are signed by elliptic curve signatures.

- *Live and Long-Term Visualization* is performed by connecting to one or more aggregation proxies, which push Transport Layer Security (TLS) protected monitoring data either directly to a live-visualization application or a database that keeps data to monitor long-term effects. The database is queried by a proof-of-concept web application, which is able to show health information by heat maps, for example. Therefore, it allows administrators to easily diagnose failures with only a few looks.

All-in-all, the monitoring system passively measures the status of the VPN in all participating nodes and uses a reverse-multicast tree to aggregate the information without burdening the VPN by the creation of additional security associations. Furthermore, no single-point-of-failure is created within the VPN components.

III. Linux-Prototype

For practical evaluation, SOLID has been implemented in a Linux environment and deployed in a testbed with more than 25 VPN nodes (see Fig. 3). The prototype offers most of SOLID's conceptual features, and uses an unmodified IPsec implementation. As an end-to-end IPsec encryption is compulsive for data traffic and the claimed IP address ranges are cryptographically authenticated during the performed Internet Key Exchange (IKE), already the prototype reaches the same level of security as a manual deployed IPsec infrastructure. Not even internal attackers can violate the confidentiality, integrity, nor authentication of the transmitted data packets. Large VPN scenarios can furthermore be evaluated by simulating SOLID utilizing OMNeT++ and INET. As most of the prototype code is reused in this case, the obtained values are highly comparable to real world tests.

The results obtained from the developed Linux prototype and simulations show that SOLID is able to increase the overall availability of VPN, while making them easier to deploy and guaranteeing the same level of conventional security as in manually configured infrastructures. As SOLID makes the distributed system more complex, the developed monitoring solution is an essential asset to help administrators to understand and protect against even complex failure situations in a distributed environment, such as sporadic failures of certain links, network partitions, uni- and bidirectional packet loss, and high delays.

IV. Planned Demonstration

For the demonstration track we plan to show:

- The functionality of the monitoring system will be introduced by *basic data forwarding*, i.e., VoIP traffic, common web services and media streaming over our VPN prototype.
- The monitoring is also used to show SOLID's advanced *resilience properties* in case of node failures, simultaneous startup of many nodes, partial connectivity loss, or network partitioning.
- To also connect data centers or head quarters with a VPN, SOLID offers the possibility to perform *load balancing* between VPN nodes. The effectiveness of this feature is also made conceivable by the monitoring approach.

The audience will observe the impact of changes on the VPN infrastructure by using our dedicated live visualization tool, which is also deployed on tablet devices. We would like to give auditors also the opportunity to control data sources, i.e., mobile live-video applications, and monitor the implications for the SOLID prototype. Furthermore, long-term effects in artificially unstable parts of the VPN are made visible by the web application.

Fig. 3. View on our deployed testbed with 25 SOLID nodes

References

[1] M. Rossberg, F. Girlich, and G. Schaefer, "Analyzing and Improving the Resistance of Overlay-Networks against Bandwidth Exhaustion Attacks," in *RNDM*, 2012.

[2] M. Rossberg, G. Schaefer, and T. Strufe, "Distributed Automatic Configuration of Complex IPsec-Infrastructures," *Journal of Network and Systems Management*, vol. 18, no. 3, pp. 300–326, 2010.

[3] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," *ACM SIGCOMM*, vol. 31, no. 4, pp. 149–160, 2001.

[4] B. Ford, "Unmanaged Internet Protocol: Taming the Edge Network Management Crisis," in *2nd Workshop on Hot Topics in Networks*, 2003.