*Roßberg, Michael; Grey, Michael; Trapp, Markus; Girlich, Franz; Schäfer, Günter:*

# Automatic VPN-Configuration with SOLID

***URN:*** urn:nbn:de:gbv:ilm1-2015210350

*Published OpenAccess:* January 2015

*Original published in:*

Praxis der Informationsverarbeitung und Kommunikation : PIK. - Berlin : de Gruyter (ISSN 1865-8342). - 36 (2013) 1, S. 42.
*DOI:* 10.1515/pik-2012-0147
*URL:* http://dx.doi.org/10.1515/pik-2012-0147
*[Visited:* 2015-01-20]

Michael Rossberg*, Michael Grey, Markus Trapp, Franz Girlich and Guenter Schaefer

# Automatic VPN-Configuration with SOLID

***Michael Rossberg:** E-Mail: michael.rossberg@tu-ilmenau.de
**Michael Grey:** E-Mail: michael.grey@tu-ilmenau.de
**Markus Trapp:** E-Mail: markus.trapp@tu-ilmenau.de
**Franz Girlich:** E-Mail: franz.girlich@tu-ilmenau.de
**Guenter Schaefer:** E-Mail: guenter.schaefer@tu-ilmenau.de

A secure communication between company sites and government agencies is often realized by virtual private networks (VPNs), connecting internal networks over untrustworthy networks, e.g., the Internet. Devices behind VPN gateways transparently exchange information over cryptographic tunnels. The deployed protocol assures the realization of confidentiality, integrity, and authentication.

Usually, the access to VPNs is organized by full-meshed topologies or *hub-and-spoke* architectures. The latter scheme is more common, as the VPN configuration is often still done manually and it allows for dynamic participants in a VPN. If *hub-and-spoke* architectures are infeasible for bottleneck and availability reasons, two reasons speak against a manual configuration: setup as well as maintenance are time- and cost-intensive and the potential for security-relevant errors increases significantly. Thus, several approaches for the self-configuration of VPN emerged [2]. However, most depend on central entities. Furthermore, none addresses tunnel-in-tunnel configurations, which are required for advanced mobility support and Denial-of-Service-resistant topologies. None of the systems assures the same level of security as a proper manual configuration.

In contrast, **S**ecure **O**ver**L**ay for **I**Psec **D**iscovery (SOLID) [3] employs a self-maintaining VPN with distinct mechanisms for discovery, routing, and topology control. Especially the latter is important as the setup of VPN tunnels is rather resource- and time-intensive (several seconds per tunnel if smartcards are used). To efficiently search for gateways within the VPN despite a low number of proactively established tunnels, SOLID deploys well-understood concepts of peer-topeer overlays [4], [1] in IPsec infrastructures.

**Basic Approach:** SOLID's topology control establishes two IPsec associations to create a ring structure. As gateways are ordered by their inner IP addresses, packet destinations can be looked up easily. Additional cross connections assure a lookup in $O(\log n)$ steps. At first glance, this does not differ much from Chord (apart from security services). However, network addresses (i.e., the keys) are not hashed due to variable subnet masks, and samples of the address space are used to plan cross connections.

**Complex Topologies:** To allow nested topologies, where gateways communicate over other gateways, SOLID embeds the ring in more complex transport networks by deploying tunnel-in-tunnel connections. In contrast to regular routing mechanisms no broadcast is used to find indirect paths. If a connection is to be established, it will be routed using the existing overlay topology first, i.e., the way the search took. This path is optimized later by local information, leading to optimal paths in common networks.

**DoS resistance:** The distributed self-configuration allows to tolerate node failures and even react to them, e.g., data can be rerouted over other gateways. To prevent DoS attacks on important VPN parts administrators may furthermore separate them from less important ones.

Prototype and simulations (cmp. Fig. 1) verify that the availability of VPN can be increased – without sacrificing conventional security objectives – by performing a *distributed self-configuration*, with means to *recover from network partitions*, to *separate important VPN parts* from less important, and to *plan proactive backup paths*. It is also possible to *reintegrate participants* over different IP addresses quickly.

**Fig. 1:** SOLID operating environment: OMNeT++ simulation (left), VPN visualization & monitoring (middle), and Linux embedded system (right).

1 B. Ford. Unmanaged Internet Protocol: Taming the Edge Network Management Crisis. In *2nd Workshop on Hot Topics in Networks*, 2003.
2 M. Rossberg and G. Schaefer. A Survey on Automatic Configuration of Virtual Private Networks. *Computer Networks*, 55: 1684–1699, 2011.
3 M. Rossberg, G. Schaefer, and T. Strufe. Distributed Automatic Configuration of Complex IPsec-infrastructures. Journal of Network and Systems Management, 18(3): 300–326, 2010.
4 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*, 31(4): 149–160, 2001.