Facilitating the Deployment of Ad-hoc Virtual Organizations with Integrated Social and Overlay Networks

Renato J. Figueiredo, P. Oscar Boykin, Pierre St. Juste, and David Wolinsky
Advanced Computing and Information Systems, University of Florida
Gainesville, FL 32611
renato@acis.ufl.edu, boykin@ece.ufl.edu, ptony82@ufl.edu, davidiw@ufl.edu

ABSTRACT

Deploying virtual organizations (VOs) is difficult for small- and medium-scale collaborations: the overheads in establishing and managing trust, and in deploying and managing computational resources distributed across multiple organizations are daunting to many potential users, presenting a barrier to entry that significantly hinders wider deployment of VOs. We advocate an approach where social networking and self-configuring overlay virtual networks are integrated in a novel way that allows simple deployment and management of ad-hoc infrastructures for VOs. There are three central principles in our approach: (1) user relationships which have been increasingly recorded in social networking systems provide the opportunity to bootstrap trust relationships; (2) connections established at a social networking layer can efficiently be mapped to the IP layer of virtual network overlays to support existing TCP/IP applications for collaboration and resource sharing while maintaining security against untrusted parties; and (3) systems integrating social and virtual networks can be self-configuring, enabling deployment of collaborative infrastructures by non-experts. We discuss motivations for this approach, describe a prototype implementation which integrates the Facebook social network and the IPOP overlay network, and discuss a use case scenario towards ad-hoc social cycle-sharing virtual Condor pools.

Categories and Subject Descriptors

C.2.4 [**Distributed systems**]

General Terms

Security, Human Factors

1. INTRODUCTION

While there are several examples of successful large-scale Virtual Organizations (VOs) in computational science and engineering, wide-spread adoption of VOs has yet to be attained. A major contributing factor to this situation is the fact that, while commodity computers have become increasingly cheaper and faster, the *management overhead* associated with deploying cross-organization, wide-area cyberinfrastructures can be larger than researchers are able to afford. This is often the case in small- and medium-scale efforts and those involving institutions which do not have extensive personnel to manage the requirements of maintaining complex computational resources.

A key management overhead associated with initiating a VO is the establishment of trust among its participating entities, typically by means of establishing a Public Key Infrastructure (PKI) certificate authority. Furthermore, a key management overhead in sustaining a VO is the system administration of distributed, cross-domain computing resources and associated software and middleware. In this paper we investigate a systems approach which can reduce both overheads through a novel approach integrating social and overlay networks into social virtual private networks (VPNs). Social VPNs are self-managing network overlays interconnecting wide-area resources where trust is established by leveraging relationships taken from social networks.

Today's social networking infrastructures allow users to manage and document their connections to others, thereby recording some aspect of the trust relationships between users. Users record their personal information in profiles and are able to discover profiles of other peers in the system. In typical social networking infrastructures, the information stored in these profiles allows users to make decisions about whether they would be willing to be linked to other peers. These infrastructures have been proven to have Web-based interfaces that are very user-friendly: surveys have shown that a large percentage of Internet users now routinely interact with social networking Web sites [9]. We argue that the user interfaces of social networking infrastructures can be leveraged to facilitate the establishment of trust among users to bootstrap ad-hoc small- and medium-scale VOs. Furthermore, the social infrastructure can be used to facilitate the exchange of cryptographic keys that can then be used for secure communication among peers through integration with IP overlay networks that establish peer-to-peer connectivity even in the presence of NATs and firewalls [5].

The goal is to securely interconnect Internet users, where end-to-end, point-to-point virtual network links are created, *automatically*, as a result of connections established through social network infrastructures. In this approach, the only configuration required from users is the creation and management of *social connections*; the configuration and maintenance of *IP network connections* is self-managing and com-

HPDC'08, June 23–27, 2008, Boston, Massachusetts, USA.

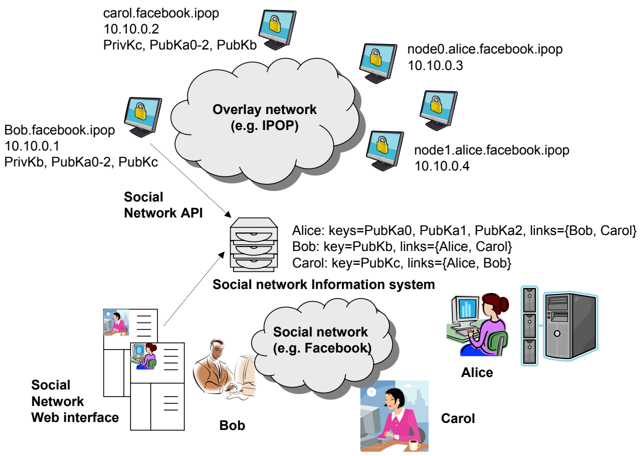

Figure 1: Integration of social and overlay networks to support ad-hoc VOs. In the example, users Alice, Bob and Carol have used a Web interface and information available from profiles to establish social links (bottom half). The information stored in the social networking infrastructure is accessed through an API by overlay virtual networking middleware to exchange keys and configure a VPN stack (e.g. IPsec) and establish secure connections among user resources. Resources are bound to virtual private IP addresses belonging to an isolated IP namespace in the overlay; virtual DNS entries are derived from social network identities and mapped to the numeric virtual IP addresses to facilitate discovery.

pletely transparent to users. Social networking enables ad-hoc VOs where no central administration is needed and users are able to establish and maintain their individual trust relationships with friendly interfaces in a way that can be accessed through APIs. Overlay networking is key to enable a system where private tunneling is setup and maintained without requiring central administration.

The social VPN functionality is thus accomplished without burdening users with the complex, error-prone configuration typically required to bring up public key and network tunneling infrastructures. Bootstrapping such a social VPN then greatly facilitates the deployment of ad-hoc VOs because it provides private, authenticated end-to-end communication channels among VO participants.

This paper is organized as follows. Section 2 describes use cases motivating social VPN infrastructure for ad-hoc VOs. Section 3 discusses an approach for architecting social VPNs and a prototype implementation. Section 4 discusses related work, and Section 5 presents concluding remarks.

2. MOTIVATIONS

The following use case scenarios illustrate cases where a social VPN facility can be used to create progressively more complex VOs from the bottom up.

Use case scenario 1: ad-hoc social VPN for collaboration among individuals: Researchers Alice, Bob and Carol, who are connected by a social networking infrastructure, decide to form a virtual organization. Alice installs a social VPN self-configuring virtual router. Upon login she is prompted to connect to her social networking site of choice. Transparently to her, the social VPN router appliance obtains a virtual IP address automatically via DHCP, generates a private/public key pair, binds Alice's user name in the social network and her virtual IP address to a self-signed certificate, and publishes her public key to the social networking site. The virtual router also periodically polls for her list of friends in the social network, and configures their X.509 certificates to be used by its IPsec stack. Bob and Carol go through the same exact process of downloading and installing the self-configuring virtual router. At the end of this procedure, the three users can communicate seamlessly and securely with each other for collaboration tasks such as conferencing, desktop sharing, data sharing, even though they might be behind different NATs and firewalls. As other users are added to an individual's social network list, they are also added to the list of social VPN participants *for that individual only* through point-to-point cryptographically authenticated/encrypted links.

Use case scenario 2: ad-hoc cycle sharing: Encouraged by their accomplishment, Alice, Bob and Carol now plan to share pools of local resources each one has in their own labs. The social VPN virtual router they download is packaged in a virtual machine appliance which also integrates a batch job scheduler which self-configures a pool of resources for opportunistic computing, such as in the Condor-based [10] Grid appliance. Each appliance instance obtains a different virtual IP address, which again is automatically and transparently self-signed with the owner's respective key, allowing multiple appliances to be bound to the same identity.

Use case scenario 3: grouping and trust delegation: Alice, Bob and Carol start interacting with other researchers in their field and the number of interested participants starts to grow. Initially adding such participants to their individual list of contacts, they eventually decide to delegate management to an entity they trust, Trent. In doing so, they configure their virtual router appliances to also trust Trent's friends list. Every participant in this VO should now seek a connection with Trent to be able to exchange keys with other participants. This scenario is now similar to a typical VO, where Trent acts as a certificate authority responsible for adding trusted users to the group according to some policy established by the group, and revoking those who may no longer be trusted. Key management is still largely transparent to end users, who now can connect to a larger number of entities in their social VPN through Trent.

3. SYSTEM ARCHITECTURE

The core technology enabling bootstrapping of a social VPN for ad-hoc VOs is an overlay network with the following features: 1) it enables automatic assignment and dynamic translation of virtual names and virtual private IPv4 addresses to hosts; 2) it supports automatic generation, exchange and discovery of peer credentials through a social networking infrastructure, and 3) it allows end-to-end authentication and encryption of all communication among trusted peers. The setup requires little configuration from users.

The overall approach of integrating social and overlay networks to bootstrap ad-hoc VOs is illustrated in Figure 1. The overlay network (top) is key to providing seamless connectivity through virtual IP tunnels among users distributed across different domains, and possibly behind different NATs, as in the IPOP overlay [5]. Social networking APIs are leveraged to exchange keys and generate private point-to-point network tunnels based on trust relationships recorded through Web interfaces (bottom). Software packaging in the form of virtual router appliances enable plug-and-play connectivity to a social VPN, and large numbers of social VPNs can multiplex a common overlay while preserving isolation.

3.1 Overlay network: routing VPN packets

The use of virtual, overlay networks has been studied in the context of dealing with challenges in connecting wide-area resources [13, 8, 14]; related approaches also have considered specialized socket libraries [11] and connection brokering [12]. The key advantage of a virtual network is application transparency. This is important for ad-hoc VOs because it allows users to run existing, unmodified TCP/IP applications, e.g. for data sharing (distributed file systems), voice/video communication, desktop sharing, and cycle sharing (with systems such as Condor [10]).

We have focused on the IPOP [5] self-configuring IP overlay network because of its support for decentralized NAT traversal, decentralized node configuration services through a distributed hash table (DHT) and the DHCP protocol, and reuse of existing PKI-based network security infrastructures (e.g. IPsec). Other overlay virtual networks could also be used towards supporting a social VPN.

3.2 Social network: bootstrapping VPN links

We have implemented a prototype social VPN that integrates the Facebook social network and the IPOP overlay, as follows[1]:

Peer discovery: We created a Facebook desktop application which authenticates a user, advertises one's keys and discovers other users who also run the social VPN software. The Facebook DataStore facility was used to store and query for user virtual IP addresses and public keys; the application automatically generates RSA private/public key pairs and uploads the public part into the social network. This information is stored in an object named IpopData as tuples (user name, virtual IP, RSA public key). Other peers discover this information by means of querying their list of friends, and the IpopData objects associated with each friend ID.

Key exchange and private tunneling: The virtual IP addresses and RSA public keys retrieved during peer discovery are used to configure IPsec for private tunneling. The RSA public keys are stored in disk and a configuration is automatically generated using the Racoon IPsec tool. In an implementation supporting point-to-point authentication, at Alice's appliance Racoon is configured to require tunneling through IPsec for outgoing packets from Alice's virtual IP to Bob's virtual IP (and incoming packets in the reverse direction) and to point to Bob's public key. Negotiation of shared session keys for the IPsec tunnel occurs on-demand with the IKE protocol when packets flow from Alice to Bob.

Name resolution: The user names retrieved in the peer discovery process above are used to populate a loop-back DNS server. This virtual DNS server maps unique names composed from Facebook IDs (e.g. as in *bob.facebook.ipop*) to a private IPOP address configured as described below. In cases where a user has multiple resources connected to the virtual network, unique names are generated by expanding the user-friendly name with resource IDs (e.g. as in *node0.alice.facebook.ipop*).

Virtual network interface setup: The social VPN configures a virtual network interface "tap" device on the host upon startup. This device is automatically configured to obtain a global virtual address in the IPOP space using the DHCP-based decentralized approach described in [7], and points to the loop-back DNS server configured to resolve social network names, as described above.

3.3 Support for x.509 based authentication

A simple extension allows the system to also support the third use case scenario in Section 2. In this case, users would configure the social network application to determine the identity of a trusted user (Trent) who acts as a certificate authority for the ad-hoc VO. Trent's social router appliance uploads the CA public key to the social network, where it becomes accessible to users. Alice and Bob now generate certificate requests which are published to the social network. Trust is established by Trent upon reviewing profiles of requesting users through the Web interface (or by additional means, if necessary). By accepting Alice and Bob to connect to him, certificates are automatically signed by Trent, cryptographically bound to Alice's and Bob's respective virtual IPs. The virtual router at Alice obtains her signed certificate by polling the social network API, and configures IPsec to use this host key to connect to others in the VO. Bob's virtual router goes through the same steps. Only when both parties have certificates signed by Trent are they able to communicate over the private VPN channel.

Management tasks can also be accomplished by Trent through mediation of a social networking infrastructure. Certificate revocation lists can be published by Trent, and the social networking application at Alice and Bob can automatically request certificate renewals when they near expiration.

3.4 Support for IP address translation

The ability to support application-transparent connectivity through IPv4 virtual addresses is crucial. However, IPv4 addresses are scarce, and without careful design, the private address space allocated by the virtual network may collide with existing physical address spaces. There are two approaches to deal with this problem. One is to use virtual network interfaces encapsulated in virtual machines, which fully decouple the overlay IP address space from the underlying network. This approach is appealing in cases where virtual machines provide an additional layer of isolation and facilitate software deployment, as in the scenarios described in [6, 8, 13].

Another approach is to support dynamic address translation, mapping a small virtual private address space (bound to a particular social network or VO) to globally unique overlay IDs. Such approach can be implemented through packet rewriting/forwarding techniques, essentially creating a network address translation within the social VPN.

3.5 Example: Cycle-sharing virtual appliance

Social VPNs can facilitate the bootstrapping of ad-hoc VOs for high throughput opportunistic cycle-sharing. One example of such system can be built around appliances that integrate overlay networks and Condor in virtual machines [4]. The baseline Condor "Grid appliance" supports autonomous configuration of virtual IP addresses and publish/discover of Condor central managers through a distributed hash table (DHT) accessible from an XML-RPC interface. It also sup-

[1] The prototype can be downloaded from grid-appliance.org

ports self-organizing flocking through the DHT, in a manner similar to the one described in [1]. Users can create independent private Condor pools (e.g. for independent VOs) on top of a single overlay by using unique IPOP namespaces. Configuring a typical pool is a simple process of deploying one appliance with a virtual floppy disk which configures it as a central manager, and deploying one or more appliances configured as workers. Workers discover managers through the DHT and advertise themselves using ClassADs.

Extending the Condor Grid appliance to integrate social networking can be accomplished as follows. User Alice allocates an IPOP unique namespace for the Condor pool, publishes the namespace ID to the social networking infrastructure, and deploys a manager appliance. Users Bob and Carol become linked to Alice and deploy worker appliances; they discover the virtual IP address of the central manager within their namespace through the DHT. Public keys are exchanged to configure IPsec at Alice, Bob and Carol such that their resources are added to the ad-hoc Condor pool through a secure, private VPN channel. Bob and Carol need to become linked if they allow jobs from each other to run on their own resources — the VPN layer blocks traffic from Bob to Carol otherwise. In a larger user base, if Alice has the role of a CA, the pair-wise linking between Bob and Carol through the social network is not necessary. Once a node is bootstrapped, its behavior (and performance) in the context of a social Condor appliance pool is not substantially different from the wide-area system described in [6, 4].

4. RELATED WORK

Several related projects have considered the use of virtual networks to facilitate the deployment and management of wide-area distributed systems, including ViNe [14], VIOLIN [8], VNET [13]. While these overlays could conceivably be used as the communication backbone of a social VPN, these projects have not addressed the use of social networks to facilitate peer discovery and key exchange. Two key aspects of the IPOP overlay which make it particularly suitable for this application are self-organization and decentralized NAT traversal. Related efforts such as Organic Grid [2] and OurGrid [3], have considered the use of P2P techniques to wide-area cycle-sharing frameworks. These systems do not leverage P2P overlays and social networking infrastructures to bootstrap ad-hoc VOs.

5. CONCLUSIONS

This paper proposes an approach to lowering the barrier to entry in the establishment of small- and medium-scale VOs. In combination with virtualization technologies and existing job schedulers, this approach facilitates the bootstrapping of secure opportunistic resource sharing pools on resources that might otherwise be idle. We have implemented a proof-of-concept prototype integrating existing social and overlay networks, virtual machines and the IPsec security framework, demonstrating the feasibility of this approach. There are several open questions that still need to be addressed: to what extent users can trust identities with the information provided by social network profiles? Can the overlay network leverage topology information from social connections to improve routing, object lookup, and multicast-based resource discovery? These questions raise interesting research topics that cross social and computer science disciplines.

6. REFERENCES

[1] A. Butt, W. Zhang, and Y. Hu. A self-organizing flock of condors. *Journal of Parallel and Distributed Computing*, 66(1):145–161, Jan 2006.

[2] A. Chakravarti, G. Baumgartner, and M. Lauria. The organic grid: Self-organizing computation on a peer-to-peer network. *IEEE Transactions on Systems, Man and Cybernetics*, 35(3), May 2005.

[3] W. Cirne, F. Brasileiro, N. Andrade, L. Costa, A. Andrade, R. Novaes, and M. Mowbray. Labs of the world, unite! *Journal of Grid COmputing*, 4(3), 2006.

[4] D. Wolinsky et al. On the design of virtual machine sandboxes for distributed computing in wide-area overlay of virtual workstations. In *Proc. Workshop on Virtualization Technologies in Distributed Computing (VTDC)*, 2006.

[5] A. Ganguly, A. Agrawal, P.O.Boykin, and R. Figueiredo. IP over P2P: Enabling self-configuring virtual IP networks for grid computing. In *Proc. IEEE Intl. Parallel and Distributed Processing Symp. (IPDPS)*, Rhodes, Greece, June 2006.

[6] A. Ganguly, A. Agrawal, P.O.Boykin, and R. Figueiredo. WOW: Self-organizing wide-area overlay networks of virtual workstations. In *Proc. Intl. Symp. on High Performance Distributed Computing (HPDC)*, Paris, France, July 2006.

[7] A. Ganguly, D. Wolinsky, P. O. Boykin, and R. Figueiredo. Decentralized dynamic host configuration in wide-area overlay networks of virtual workstations. In *Proc. Workshop on Large-scale, Volatile Desktop Grids (PCGrid).*

[8] X. Jiang and D. Xu. Violin: Virtual internetworking on overlay infrastructure. In *Proc. 2nd Intl. Symp. on Parallel and Distributed Processing and Applications (ISPA)*, pages 937–946, 2004.

[9] M. Rankin Macgill A. Smith A. Lenhart, A. Madden. Teens and social media: The use of social media gains a greater foothold in teen life as email continues to lose its luster. Pew Internet and American Life Project, Dec 2007.

[10] M. Litzkow, M. Livny, and M. Mutka. Condor - a hunter of idle workstations. In *Proc. 8th IEEE Intl. Conf. on Distributed Computing Systems (ICDCS)*, June 1988.

[11] J. Maassen and H. Bal. Smartsockets: Solving the connectivity problems in grid computing. In *Proc. IEEE International Symposium on High Performance Distributed Computing (HPDC)*, Monterey, CA, 2007.

[12] S. Son and M. Livny. Recovering internet symmetry in distributed computing. In *Proc. 3rd Intl. Symp. on Cluster Computing and the Grid*, May 2003.

[13] A. Sundararaj, A. Gupta, and P. Dinda. Dynamic topology adaptation of virtual networks of virtual machines. In *Proc. 7th Workshop on Languages, Compilers and Run-time Support for Scalable Systems (LCR)*, Oct. 2004.

[14] M. Tsugawa and J. A. B. Fortes. A virtual network (vine) architecture for grid computing. In *Proc. IEEE Intl. Par. Distrib. Proc. Symp. (IPDPS)*, Rhodes, Greece, June 2006.