15th ICCRTS

"The Evolution of C2"

Experiments with Web services at Combined Endeavor

Paper ID 002

Topic 5: Experimentation and Analysis, Topic 2: Networks and Networking, Topic 9: C2 Architectures and Technologies

Frank T. Johnsen and Trude Hafsøe

Point of contact:
Frank T. Johnsen
Norwegian Defence Research Establishment (FFI)
P.O. Box 25, NO-2027 Kjeller, Norway
+4763807960
frank-trethan.johnsen@ffi.no

Abstract

Web services technology has been identified as a key enabling technology for NATO NEC. The benefits of this technology is that it provides loose coupling of systems (which facilitates interconnecting existing systems) and that it is based on standards (which is important for achieving interoperability). Web services have gained widespread use on the Internet, but there are challenges that need to be addressed when using this technology in military networks, particularly disadvantaged grids. At Combined Endeavor we have demonstrated that we can use Web services across heterogeneous networks, and that we can utilize this technology not only in deployed but also in mobile tactical networks. This paper describes our experiments with Web services at Combined Endeavor in 2009. In previous experiments we have shown that it is possible to invoke Web services in military networks. At CE, we wanted to explore the use of Web services technology in a combined operation, by employing service discovery and invocation both in and across heterogeneous military networks.

Introduction

One of the main goals of Network Enabled Capability (NEC) is to increase mission effectiveness by interconnecting military entities. Sharing information between decision-makers can help guide them towards making the right decisions at the right time, and a common information infrastructure is needed to facilitate sharing of relevant information across system and national boundaries. The NATO NEC feasibility study (NNEC FS) [1] envisions the concept of a Service-Oriented Architecture (SOA) to become pervasive in this information infrastructure. In a SOA, networked resources are made available to others as a collection of services, often implemented by using a technology called Web services [2]. Current Web services solutions are designed for Internet-type networks, but our previous research (see the section below) has shown that it is feasible to invoke known Web services in military tactical networks when using different optimization techniques. This means that provided you know the invocation address of a Web service, the so-called *address location* of the service, then that service can be used. We have successfully demonstrated that Web services can be invoked in disadvantaged grids in earlier experiments (e.g., at NATO CWID [3] and in the national Multinett II exercise [10]).

In a highly dynamic environment, such as a military mobile ad hoc network (MANET), being able to locate Web services becomes a major challenge [9]. The process of identifying a service, known as *service discovery*, is an important part of any SOA, but it is particularly challenging in dynamic environments. A service discovery architecture for such an environment should offer a *complete and up-to-date picture* of the services available at any given point in time. Responses to queries should *mirror the current state in the service network* and should not advertise services that are no longer present in the network (i.e., reflect so-called service *liveness*) [14].

Combined Endeavor (CE) is an annual communications exercise. CE 2009 was the 15th year for this multinational event. It was the first time three separate locations were used for experiments, and also the first time that a Partnership for Peace Nation, Bosnia-Herzegovina, was used as the main site. The other exercise sites included the Netherlands and Denmark.

At CE 2009 we successfully demonstrated dynamic service discovery *in and across* heterogeneous tactical networks. Following discovery, we were able to invoke the Web services. Thus, we demonstrated that we could use Web services as a middleware across network and national

boundaries in an interoperable and agile operation. Our main partner at CE was the NATO C3 Agency (NC3A), and the experiments described in this paper were performed at the exercise site in the Netherlands.

Previous research

In our previous work we have investigated several aspects of adapting Web services technology for use in military networks.

At NATO CWID in 2007 (see [3]) we experimented with Web services in an emulated disadvantaged grid. We SOA-enabled a legacy system with Web services, allowing it to provide NFFI tracks to a central HQ which could then visualize the track information and build a COP. The central HQ was connected to our experiment partners, where we used XML security labeling and an XML guard to secure the communications. In other words, we used Web services for point-to-point connections between different end systems. In these experiments we found that using optimizations such as XML compression and optimized transport protocols, as well as store-and-forward functionality was a necessity in order to enable Web services in disadvantaged grids [4].

We have surveyed central Web services standards and specifications, and found that Web services are well suited for building not only traditional "pull" type systems (i.e., request-response operations), but also "push" type systems (i.e., event driven operations). We discuss the two specifications, WS-Notification and WS-Eventing, and their importance for military systems in [5]. Also, we discuss proxy servers (i.e., intermediate nodes between clients and services) and that they can be used to provide added value operations, such as compression, content filtering, and so on.

The most common way of implementing Web services is by using HTTP/TCP for transport. Since TCP does not work well in most disadvantaged grids due to low data rates, varying throughput, disruptions and high error rates, we have investigated and shown that it is possible to use military message handling systems (STANAG 4406) as a carrier for the Web services protocol SOAP [6].

Using the XML version of NATO Friendly Force Information (NFFI) STANAG 5527 as a case study, we have investigated optimizing the *information overhead* by performing application specific content filtering [7]. Content filtering reduces the total information overhead, leading to less information that needs to be transmitted across the network. In addition, we have investigated various ways to reduce the XML overhead, by comparing the compression performance of several algorithms [8]. We found that a generic compression algorithm like GZIP compresses XML well, but that the emerging W3C standard for XML compression, Efficient XML (EFX), has an edge over GZIP. Being an XML conscious compression technique, EFX uses knowledge of the XML structure to perform its compression, thus having an advantage over the generic algorithms. No matter which algorithm we use, we found that compression in some form should definitely be employed, since it significantly reduces the size of XML documents.

Using Web services as a means of integrating stove pipe systems is a requirement of NNEC, and we have attempted that in the "Multinett II" joint national experiment (see [10]) in 2008. There we were able to interconnect systems from the navy and air force, in order to achieve a cooperative electronic support measures operation. Using Web services as a means of integration and interoperability, we could show the added value of employing Web services in military operations. Also, we

demonstrated parts of the standardized XML security mechanisms (e.g., XML signatures). We also identified some challenges related to using Web services as a middleware, in that we found that standardized Web services discovery mechanisms do not necessarily function well in disadvantaged grids (we attempted to use WS-Discovery, but found that it generated too much traffic in our network, flooding buffers and disrupting other traffic).

Service discovery is important in dynamic environments because services can come and go, and we need to know which services are available at any time. In [9] we discuss the requirements and challenges of service discovery in different military networks. We conclude that due to the diversity of the networking technologies used in military networks, one mechanism cannot be used in all networks. We need a toolkit of different mechanisms, where the mechanism that is best suited is used at any time. By doing this (for example by using specially optimized solutions such as our experimental *Service Advertisements in MANETs* (SAM) (see [11]) in disadvantaged grids) we can solve the problem of service discovery *in* military networks. However, interoperability is a key concern, so there is also a need for pervasive service discovery *across* heterogeneous networks. We have investigated pervasive service discovery in [13], where we conclude that using *gateways* for interoperability is the simplest and most cost-efficient means to achieve the needed protocol interoperability. The gateways must be placed in the connection points between heterogeneous networks (i.e., the so-called *interoperability points* that the NNEC FS discusses, see [1]), and provide transparent service discovery protocol interoperability.

A key concern when adopting NNEC is to keep costs down by using COTS technology when possible. Some of the techniques discussed above break Web services standards, but are necessary to get Web services to work in disadvantaged grids. By implementing the optimizations in proxies, we can continue to implement and use COTS technology in clients and servers. The proxies intercept standard Web services and perform the necessary optimizations on inter-proxy traffic. Proxy concepts are discussed in [5], and our Delay and Disruption tolerant SOAP Proxy (DSProxy) prototype is presented in [12].

Experiment motivation

In previous experiments we have shown that it is possible to invoke Web services *in* military networks. At CE, we wanted to explore the use of Web services technology in a combined operation, by employing service discovery and invocation both *in and across* heterogeneous military networks. We wanted to employ Web services standards as much as possible, augmenting the system with proprietary, experimental solutions only where necessary. We wanted to test our prototype DSProxy, which can enable COTS Web services clients and services to operate across heterogeneous environments. By deploying the proxy software locally in each network node, the proxy can intercept the standard Web services invocation locally. This means that SOAP over HTTP and TCP is used between client and proxy, and between proxy and server. However, the proxy supports compression, multiple transport protocols and adds delay and disruption tolerance, meaning that the communication *between* the proxies can be performed across heterogeneous networks. A simple deployment like this with locally deployed proxies communicating across heterogeneous networks is shown in Figure 1.

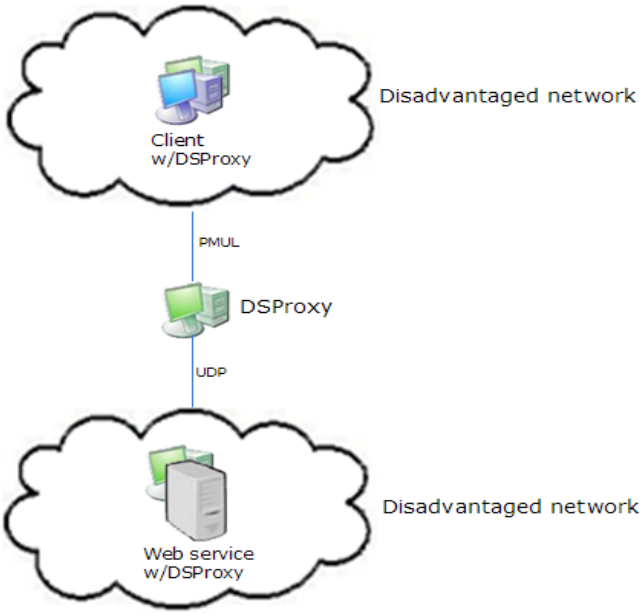

Figure 1 Locally deployed proxy configuration (from our paper [12])

Also, we wanted to achieve pervasive service discovery. In previous experiments we have performed the service discovery at *design-time* (i.e., the service endpoints used in the experiments have been hardcoded and static in the applications). This way of using Web services is common in civil applications, where the services and the network infrastructure are stable. In tactical networks, however, there is a need for *run-time* discovery, since the dynamic nature of the system means that services are transient. After a theoretical evaluation of the suitability of different service discovery mechanisms (see [9]), we categorized different existing service discovery mechanisms according to their suitability for use in military networks. This is illustrated in Figure 2. The slight overlap between the technologies shown indicates that there is not necessarily a hard limit between the operational levels when it comes to using service discovery technology. Instead, it shows which level will most likely benefit from using techniques in these categories based on the number of services and users, and also the characteristics of the communications technology which is typically used at this level. That it is possible to use Web services standards at the strategic level where the infrastructure is based on Internet technologies is obvious. Web services were designed for use in such networks. However, we wanted to experiment with this technology at the tactical level, both for deployed and mobile networks.

We wanted to explore two cases:

- First, we wanted to show pervasive use of Web services (i.e., discovery and invocation) across network and national boundaries. We used a traditional setup, where direct communication between the two MANETs was not possible. Instead, all communication had to go via the interoperability point between the two HQs. Interoperability between the nations was provided by using TACOMS (communications standard for joint operations, see http://www.tacomspost2000.org/) and the common CE backbone.
- Second, we wanted to try another use case: That of direct interoperability between the two MANETs. This required the use of another interoperability point to connect the two different technologies together.

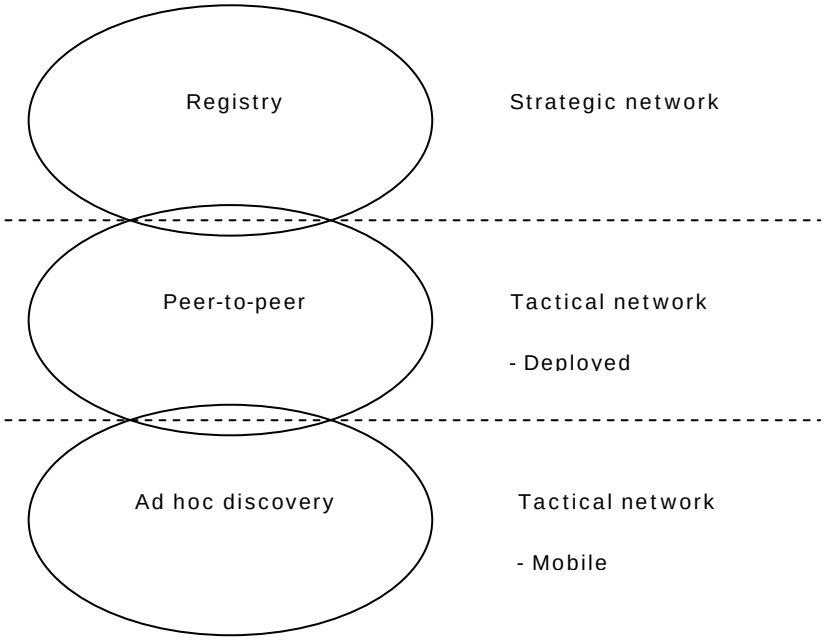

Figure 2 Suggested service discovery mechanisms for each operational level (from our paper [9])

Experiment setup

Our first experiment setup at CE was as shown in Figure 3. We had two MANETs. In the Norwegian MANET, the nodes were vehicles equipped with a tactical radio. These units were mobile, and were out and about reporting incidents (e.g., text and images) back to the base.

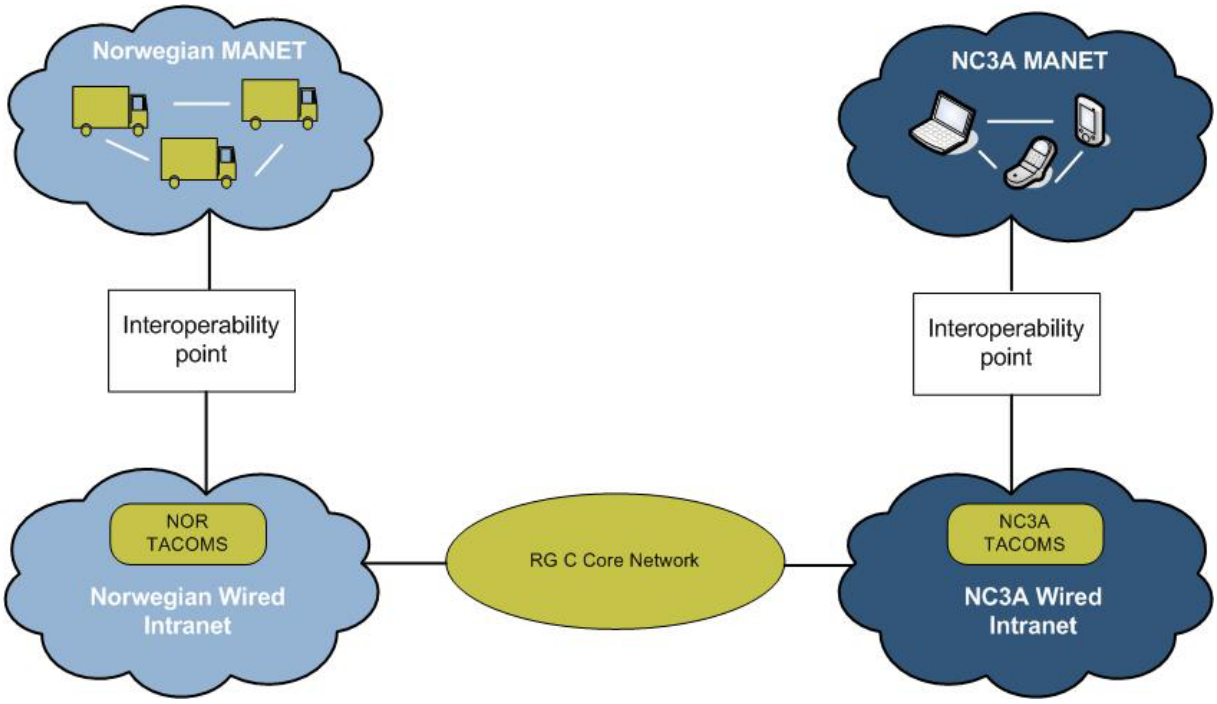

Figure 3 First experiment series setup

To make Web services work in the MANET we utilized techniques such as store-and-forward SOAP and data compression, implemented in the DSProxy prototype. Furthermore, we used a service discovery mechanism specially tailored for MANETs to discover the available services. NC3A used similar techniques in their MANET. In the Norwegian base we were able to use unmodified Web services, and we could also use the standardized discovery mechanisms such as ebXML and WS-Discovery there. The ebXML registry was connected to the NATO Metadata Registry and Repository (NMRR) in a registry federation through the RG C core network. Between the Norwegian MANET and Norwegian HQ we had an experimental gateway featuring transparent service discovery protocol translation for interoperability. In this case the gateway was responsible for interoperability between these two operational levels.

Our second experiment setup was similar, but then the connection between the two HQs was removed. The interoperability between the MANETs was done through direct communication via a second service discovery gateway, as shown in Figure 4.

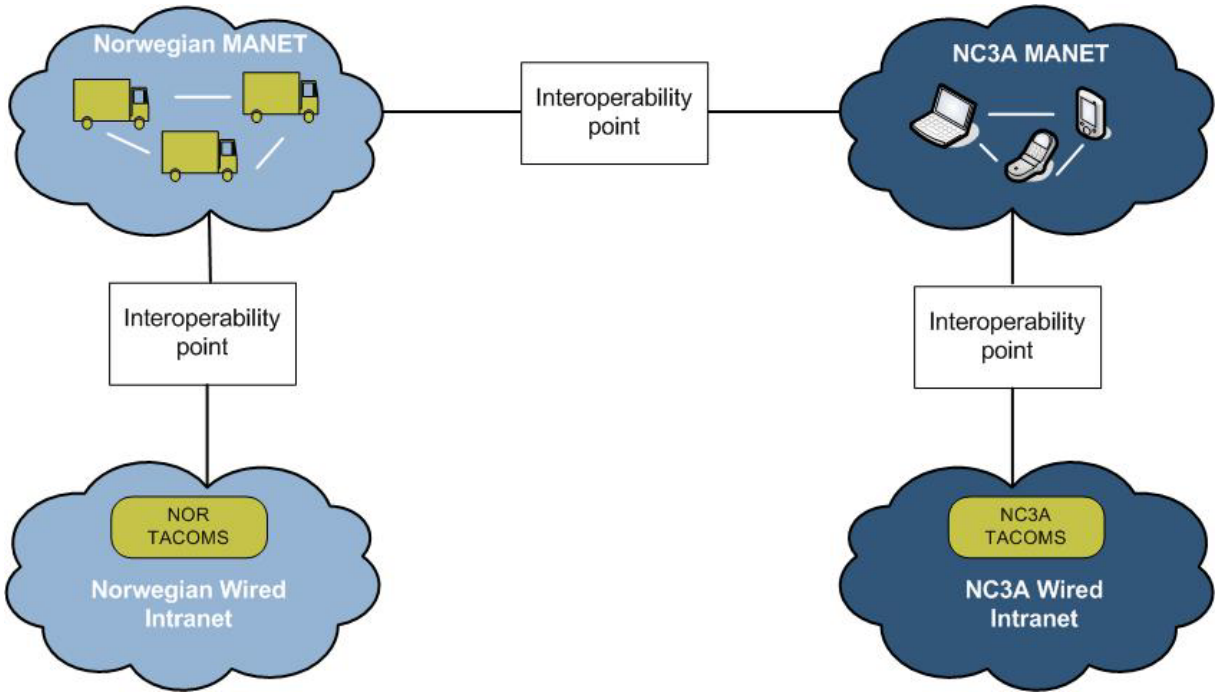

Figure 4 Severed connection between HQs, interoperability directly between the MANETs

In this case, a service discovery gateway was deployed in the interoperability point between the Norwegian MANET and the NC3A MANET as well.

Software

At CE we had no strategic network (see the levels in Figure 2 compared to the CE setup in Figure 3); we had a setup with two deployed HQs (i.e., tactical deployed networks) and two tactical mobile networks. To address the different characteristics of the networks, we chose to use registries in our HQs, and interconnect them using the registries' federation mechanism. This allowed us to perform federated searches, meaning that querying your local registry would propagate the query also to the other registries in the federation. Setting up a registry federation requires the registries to be interoperable. For Web services, two competing registry standards exist: UDDI and ebXML. Since NMRR is built on ebXML, we chose to use ebXML in the Norwegian HQ to ease interoperability. We

used the open source reference implementation of ebXML v3.0, the so-called “Omar”, which is available from http://ebxmlrr.sourceforge.net/.

Registries were created for use in large, fixed infrastructure networks. They are not suitable for use in MANETs, due to the dynamic nature of such networks. MANETs, being characterized by mobile units and unstable links, are prone to network partitioning where not all nodes can communicate with each other all the time. In such networks, you can encounter problems with service *liveness* and service *availability*:

- The liveness problem (see Figure 5) occurs when a service has been published in a registry, but the service has become unavailable. In this case, a client can still look up the service in the registry, but the service cannot be reached. No matter how many times the service discovery is performed – the result is still the same. This occurs because registries require you to actively register and de-register services to keep them up to date.
- The availability problem (see Figure 6) occurs when the registry becomes unavailable. For example, if a network is partitioned in such a way that the registry is in one partition, and the service and client are both in the other partition. If this occurs, then the client will be unable to look up any services at all, since it cannot contact the registry. Thus, even if the service a client needs is available and present in the same partition as the client, there is no way to discover it.

This means that in dynamic networks where partitions can occur, such as in tactical mobile networks, one should preferably use other service discovery mechanisms that address these issues. Tactical mobile networks usually contain a few but highly mobile participating nodes. This means that it is feasible to use fully decentralized service discovery mechanisms in such networks. A fully decentralized mechanism addresses the *availability* problem by distributing the same information about services to all the nodes that it can reach. If the mechanism is coupled with a lease mechanism or just lets service advertisements time out from its cache, then it can also address the *liveness* problem in that there is no need to actively de-register unavailable services any more – the mechanism removes such stale information itself.

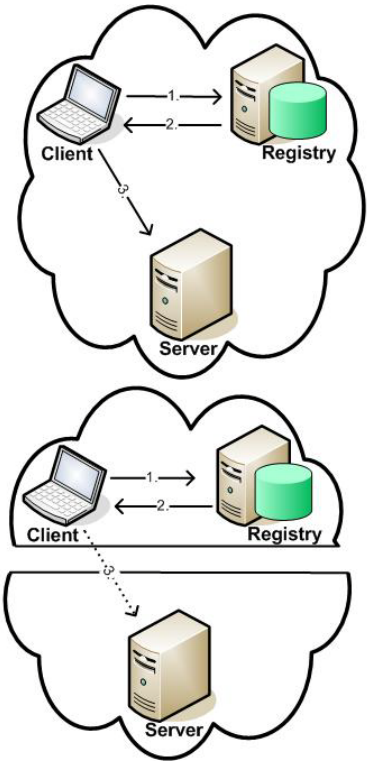

Figure 5 The liveness problem

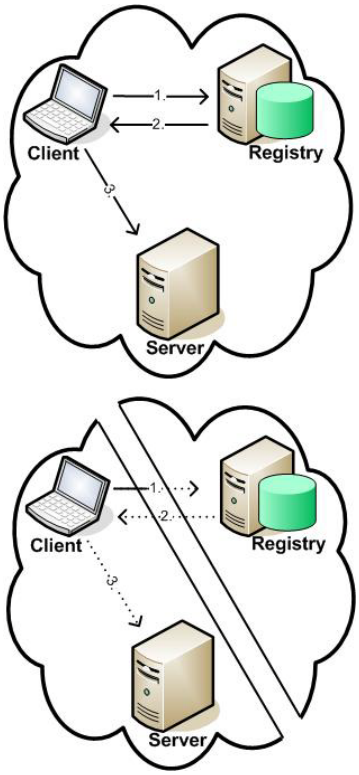

Figure 6 The availability problem

We summarize the details of our choice of discovery mechanisms in the Norwegian network in the table below:

	Web services Dynamic Discovery (WS-Discovery)	**Electronic Business XML (ebXML)**	**Service Advertisements in MANETs (SAM)**
Category	Decentralized LAN mechanism	Centralized WAN mechanism	Decentralized multihop MANET mechanism
Service descriptions	Port types and service names	WSDL and optional metadata	WSDL and optional position and metadata
Standardized	Yes	Yes	No, experimental
Operation	Fully decentralized or multicast suppression using central discovery proxy.	Centralized registry and repository. Offers federated queries by forwarding queries to other registries.	Fully decentralized using IP multicast.
Suitable for Web services discovery in highly dynamic environments	Yes, but only when running in decentralized mode.	No	Yes
Suitable for disadvantaged grids	No	No	Yes
Application in the Norwegian network at Combined Endeavor	This Web services discovery mechanism is tailored for LAN usage, and is well suited for use in the deployed HQ. It provides integration with the Norwegian registry by wrapping it in a discovery proxy.	This registry is used in the Norwegian HQ. It contains all the static Web services offered from the HQ. By connecting the registry to the NMRR we can perform interoperable federated queries in the coalition.	The mechanism is tailored for disadvantaged grids, where it provides optimized Web services discovery in a fully decentralized manner. We use this mechanism in the Norwegian tactical MANET.

In the Web services world, WS-Discovery is a standard for decentralized Web services discovery. Previously, we have attempted to use WS-Discovery in a disadvantaged grid, and found that it was unsuitable there since it generated too much traffic in the network (see our paper [10]). The WS-Discovery implementation we used in that experiment was the open source implementation available from http://code.google.com/p/java-ws-discovery/. Since then we have developed an experimental solution for service advertisements in MANETs (or *SAM* for short), that can be used instead of WS-Discovery. It supports Web services discovery in a fully decentralized manner, with support for disseminating position information along with the service advertisements. Based on NFFI, this positioning information can be collected and assembled into full NFFI tracks in the HQ, providing the added value of blue force tracking together with disseminating the Web services information. SAM uses techniques such as data compression, caching and timeouts to minimize the data rate requirements while at the same time addressing the liveness and availability problems (see our paper [11] for further information about SAM). By using this experimental mechanism, we can achieve Web services discovery in our MANET. However, since the mechanism is experimental, it is not interoperable with any of the standardized mechanisms. To address this, we implemented service discovery gateways (see our paper [13]) that we placed in the interoperability points between the networks:

- The gateway between the Norwegian MANET and the Norwegian HQ translated between SAM and the standardized WS-Discovery mechanism. This allowed us to be interoperable with a standard, which through a so-called discovery proxy (discussed in the WS-Discovery standard), again could provide further integration with the ebXML registry in our HQ.
- The gateway between the Norwegian MANET and the NC3A MANET translated between SAM and the mechanism NC3A used in their MANET, an experimental peer-to-peer based technology called Service Oriented Peers (SOP) (see [15] and [16] for further details).

Hardware

The Norwegian and NC3A HQs used civil technology: Commercial off-the-shelf switches, network cables, and routers. Forming two separate networks deployed in two separate tents at the experiment location, these networks were interconnected through the RG C backbone using TACOMS nodes.

The Norwegian MANET consisted of WM600 tactical radios capable of forming a multi-hop MANET. The NC3A used Breadcrumb radios from Rajant, which basically are ruggedized components using civil 802.11b/g technology for communications. Both technologies are IP-enabled, and can carry Web services traffic. However, the radios are not compatible on the air, since WM600s typically are configured to use a military frequency, whereas 802.11b/g uses the civil 2.4GHz ISM band. WM600 supports IP multicast, meaning that we can use SAM for discovery. The Breadcrumbs do not support multicast, but NC3A's SOP relies on unicast to a central node or set of nodes (JXTA P2P mechanism).

Experiment execution

Since the MANETs are not compatible on the air, we had to use an interoperability point. At CE, we solved this by deploying both a WM600 and a Breadcrumb in the Norwegian HQ, where the radios were both connected to a laptop computer. This computer functioned as a router between the two MANETs. This gateway could basically be placed anywhere, it could even have been mobile joining

one of the mobile nodes in a MANET. However, for convenience (i.e., continuous power supply for our laptop and radios) we chose to co-locate this gateway with our HQ at CE. Following the same principle we connected our Norwegian MANET to the HQ. Both the router laptops were running our experimental software: The DSProxy for delay tolerant invocation, and the service discovery gateway software for transparent pervasive service discovery.

The experiment was performed in two iterations:

- First (see Figure 3), we used the backbone to communicate between the Norwegian HQ and the NC3A's HQ. There was no connection between the two MANETs in the first iteration.
- Second (see Figure 4), we used an interoperability point to facilitate communications directly between the two MANETs. We severed the connection between the two HQs in this experiment.

In both iterations, the participants aimed to solve a simple mission: The NC3A units were scouting an area, and reported an observation into JOCWatch, which was an incident report Web service and database in the NC3A HQ. Upon receiving this incident in the HQ, the NC3A observed the blue force tracking system, and since the Norwegian units were in the area, they were notified via chat (made with the DSProxy), and told to investigate. The Norwegian units could then discover the JOCWatch service, connect to it, and download all the details regarding the incident (e.g., description and position). Following this the units would converge on the position, providing images from the area to the NC3A HQ via a publish/subscribe Web service. A publish/subscribe Web service is discovered in the same manner as a regular request/response service, in that you discover the point to subscribe to. After this point has been discovered you make a subscription, and any new information pushed by this service (in this case, new images) will be delivered to you.

The subscriptions were set up when the Norwegian units were dispatched, and continued for the duration of the experiment. The chat service was also implemented as a publish/subscribe service, where subscriptions were set up prior to starting the experiment. When the units reached the position, they would report back to base via chat. Following chat coordination, the units would return home to base.

The Norwegian units, being in a MANET using SAM for service discovery, were able to provide a periodic update of available services and unit positions. Since SAM provides NFFI position data, we could assemble these positions in the Norwegian HQ and offer an NFFI blue force tracking Web service to the NC3A. The NC3A units, using SOP for service discovery, exposed no positioning information, and we were unable to see their whereabouts during the experiment.

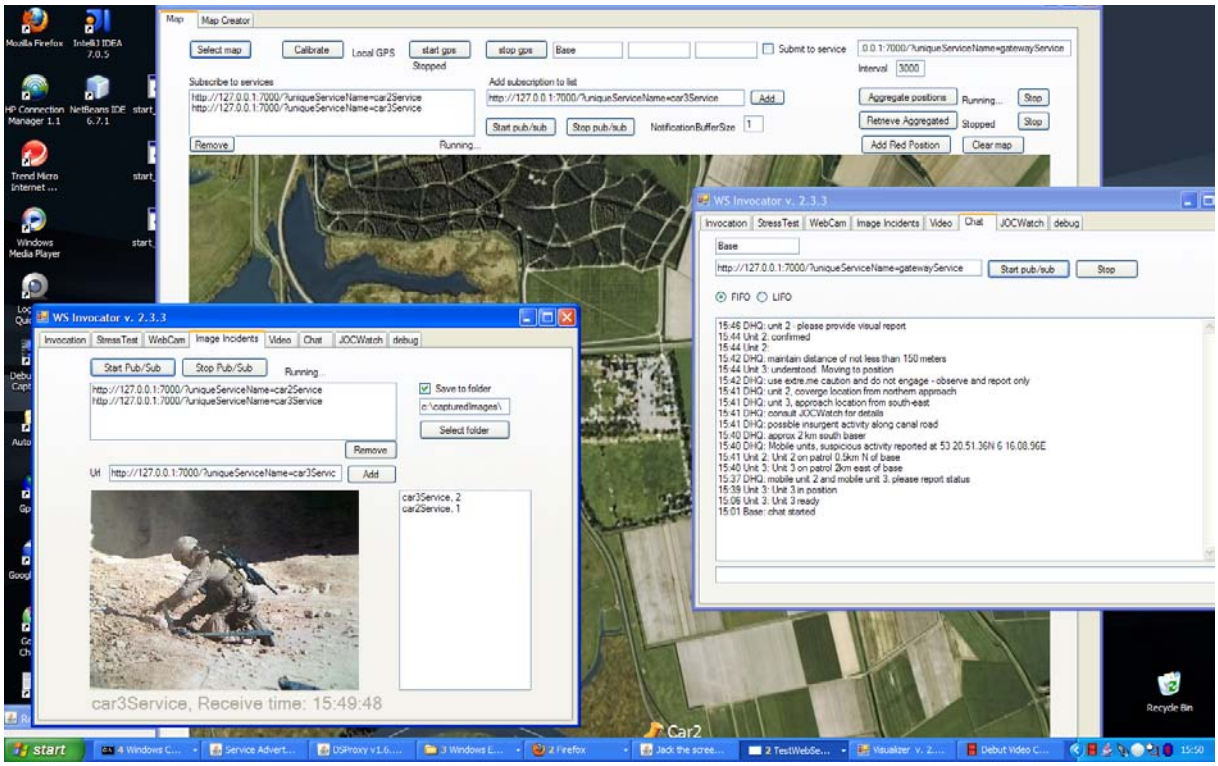

Figure 7 Screenshot from the Norwegian HQ

The experiment was a success; our software functioned as it was supposed to during both of the two different experiment runs. Figure 7 shows a screenshot from the Norwegian HQ, where you can see the chat window (right), the client for the image publish/subscribe service (left), and the simple blue force tracking system (background).

Lessons learned

Lower bandwidth may not be a big issue for Web services, but unreliable connectivity is a problem. This can be mitigated by store-and-forward techniques such as implemented in the DSProxy. However, with the potential for an unstable network, Web services are not suitable for real-time data.

We have seen that service discovery is possible in and across heterogeneous networks. However, by using a transparent gateway to translate between discovery protocols you may lose some service information going from one network to the other. For example, SAM supports both service and position information, but WS-Discovery supports only service information. This meant that our NFFI tracks had to be assembled and built by the gateway, since it was the point receiving the position information. The NFFI tracks could then be exposed as a Web service. The important thing about using gateways for interoperability is that it is sufficient to know the interface used by another network; you do not need to know the functionality details. This was the case with SOP, where we were able to extract service information, despite being unaware of the NC3A's network topology and how SOP it was deployed in their network.

We noticed some issues when using Omar, the open source ebXML reference implementation: First, it was not easy to install. You need several old Java libraries to get it to work, since it is incompatible

with some of the newer ones. Second, you have to use Sun's own Java as other implementations, such as OpenJDK do not support all the functionality (e.g. security) required by Omar. Third, there were issues configuring Omar properly. Omar comes with two user interfaces, one Web interface and one Java interface. The Java GUI and the Web GUI support different operation sets. In practice, you need to use both. However, neither GUI fully supports the configuration of federations. In order to make federated queries work we had to update the repository database manually. This is both complex and time consuming, but if you want to use Omar you have to either live with it or write a new GUI that supports all the necessary functionality. The NC3A had remedied this situation by creating the NMRR – their GUI to ebXML.

Conclusion

The results show that in disruptive environments, a specially tailored discovery mechanism must be used to overcome the problems with liveness and availability. Furthermore, a store-and-forward mechanism must be employed. When our prototype proxy was not used to support Web services invocation, our applications failed.

Our use of registries shows that they can be employed in the deployed HQ, and they can also be used in a federation between HQs. The NC3A has shown that P2P can be employed (i.e., the SOP in their network), and while this technology is mostly suitable in large fairly static networks, it can also be employed to some degree in dynamic networks. In highly dynamic networks decentralized mechanisms should preferably be used, since they address the aspect of service availability and liveness. We addressed these issues by using our experimental SAM mechanism in our MANET. Interoperability between heterogeneous networks and mechanisms can be achieved by

- Using service discovery gateways which translate between discovery protocols.
- Deploying proxies that optimize service invocation across the networks.

The issues we encountered with the ebXML reference implementation clearly show that while standards are important for *interoperability*, the maturity of the available products is equally important for system *usability*.

For further information about our experiments at CE, see our experiment report [17].

References

[1] P. Bartolomasi, T. Buckman, A. Campbell, J. Grainger, J. Mahaffey, R. Marchand, O. Kruidhof, C. Shawcross, and K. Veum. *"NATO network enabled capability feasibility study"*, Version 2.0, October 2005.

[2] Thomas Erl. *"Service-Oriented Architecture - A Field Guide to Integrating XML and Web services"*, Prentice hall, ISBN 0-13-142898-5, 2004.

[3] Raymond Haakseth, Tommy Gagnes, Dinko Hadzic, Trude Hafsøe, Frank T. Johnsen, Ketil Lund, and Bård Karsten Reitan. *"SOA - cross domain and disadvantaged grids - NATO CWID 2007"*, FFI report 2007/02301, ISBN 978-82-464-1272-6, 2007.

[4] Ketil Lund, Anders Eggen, Dinko Hadzic, Trude Hafsøe, and Frank T. Johnsen. *"Using Web services to Realize Service Oriented Architecture in Military Communication Networks"*, IEEE Communications Magazine, October 2007.

[5] Trude Hafsøe, Frank T. Johnsen, Ketil Lund, and Anders Eggen. *"Adapting Web services for limited bandwidth tactical networks"*, 12th International Command and Control Research and Technology Symposium (ICCRTS), Newport, RI, USA, 2007.

[6] Frank T. Johnsen, Anders Eggen, Trude Hafsøe, and Ketil Lund. *"Utilizing military message handling systems as a transport mechanism for SOA in military tactical networks"*, NATO IST-083 Symposium on Military Communications with a special focus on Tactical Communications for Network Centric Operations, Prague, Czech republic, April 2008.

[7] Trude Hafsøe and Frank T. Johnsen. *"Reducing network load through intelligent content filtering"*, 13th International Command and Control Research and Technology Symposium (ICCRTS), Seattle, WA, USA, June 2008.

[8] Frank T. Johnsen, and Trude Hafsøe. *"Using NFFI Web services on the tactical level: An evaluation of compression techniques"*, 13th International Command and Control Research and Technology Symposium (ICCRTS), Seattle, WA, USA, June 2008.

[9] Frank T. Johnsen, Trude Hafsøe, and Magnus Skjegstad. *"Web services and Service Discovery in Military Networks"*, 14th International Command and Control Research and Technology Symposium (ICCRTS), Washington DC, USA, June 2009.

[10] Trude Hafsøe, Frank T. Johnsen, Nils A. Nordbotten, and Espen Skjervold. *"Using Web services and XML Security to Increase Agility in an Operational Experiment featuring Cooperative ESM Operations"*, 14th International Command and Control Research and Technology Symposium (ICCRTS), Washington DC, USA, June 2009.

[11] Frank T. Johnsen. *"An NFFI-based Web service discovery mechanism for tactical networks"*, Military Communications and Information Systems Conference (MCC 2009), Prague, Czech Republic, September 2009.

[12] Espen Skjervold, Trude Hafsøe, Frank T. Johnsen, and Ketil Lund. *"Delay and disruption tolerant Web services for heterogeneous networks"*, IEEE MILCOM, Boston, MA, USA, October 2009.

[13] Frank T. Johnsen, Joakim Flathagen, and Trude Hafsøe. *"Pervasive Service Discovery across Heterogeneous Tactical Networks"*, IEEE MILCOM, Boston, MA, USA, October 2009.

[14] Tommy Gagnes. *"Assessing Dynamic Service Discovery in the Network Centric Battlefield", IEEE MILCOM, Orlando, FL, USA, October 2007.*

[15] M. Amoretti et al, *"SP2A: a Service-oriented Framework for P2P-based Grids"*, In proceedings of the 3rd International Workshop on Middleware for Grid Computing (MGC05), Grenoble, France, 2005

[16] D. Marco-Mompel, *"SERVICE ORIENTED PEER PROTOTYPE FOR MOBILE USERS"*, NC3A Technical Note Draft under project SPW001495, November 2007.

[17] Frank T. Johnsen, Joakim Flathagen, Trude Hafsøe, Magnus Skjegstad, and Nanda Kol, "Interoperable Service Discovery: Experiments at Combined Endeavor 2009", FFI Report 2009/01934, November 2009.