Resilience and Survivability for future networking: framework, mechanisms, and experimental evaluation

Deliverable number	6.2a
Deliverable name	Links between research and experimentation (interim)
WP number	6
Delivery date	28/02/2009
Date of Preparation	9/4/2009
Editor	Paul Smith(ULANC)
Contributor(s)	Paul Smith (ULANC), Merkouris Karaliopoulos (ETHZ)
Internal reviewer	David Hutchison(ULANC)

Summary

The deliverable is one of the project "light" deliverables requested by ESC in the context of the project's commitment to close interaction with the FIREworks Coordination Action. The deliverable aims at providing inputs to FIREworks for the compilation of a deliverable on links between research and experimentation, which will aggregate contributions from all project running under the FIRE initiative.

The deliverable explains how experimentation will be applied to validate our research findings within the ResumeNet project. In short, the ResumeNet work package structure is such that experimentation results feed into a core task that runs for the duration of the project. This task is intended to consolidate these results, and produce a number of deliverables throughout its lifetime that describe best practices and strategies for building resilient networked systems. To conduct our experiments, we will use a number of bespoke testbeds. While there are no overt plans to integrate these into wider testbeds, the project is open to suggestions from the FIREworks coordination action regarding this matter. Finally, the deliverable discusses open issues relating to testbed sustainability and desired features.

Experimentation within the project does not officially start until M18, therefore the discussion presented here describes the current thinking and should be viewed as ongoing work; aspects of it could be changed in due course.

Contents

1 Scope of needed experimentation

The technical work package structure of the ResumeNet project is shown in Figure 1. Tasks within WP1 will look at the fundamental aspects of building resilient networked systems – metrics, policies, cross-layering, and understanding how systems are challenged. Furthermore, within WP1, there is a task on informing a resilience framework, which is used to consolidate the learning from the project and inform activities throughout the other WPs in the project. Research on algorithms and mechanisms for resilience will be carried out in WPs 2 and 3, which will look at network- and service-level resilience, respectively. For example, in WP2, algorithms for resilient transport protocols will be developed, and in WP3 the use of virtualization for resilience will be studied. These tasks will build upon the fundamental work that is carried out in WP1. For example, we will need to understand how to effect a particular remedy to a challenge has been by understanding appropriate metrics, which are defined in WP1. A monitoring system that uses cross-layer information, developed in WP1, will be fundamental the detection and remediation work carried out in these two WPs.

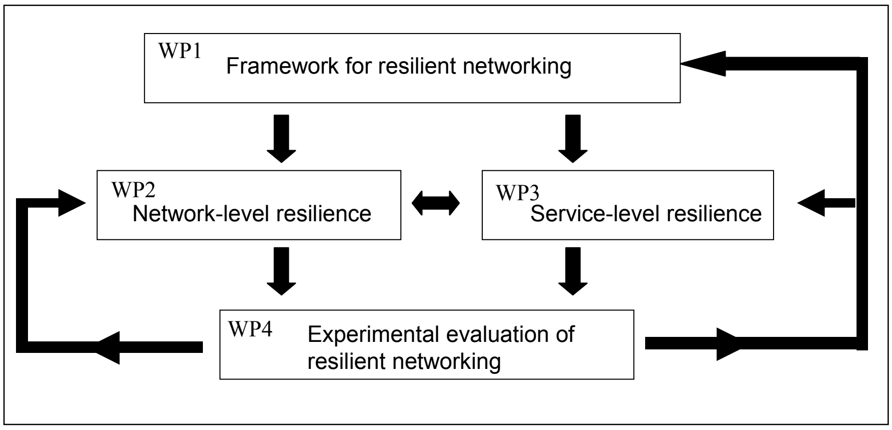

Figure 1: The ResumeNet project work package structure

The project will use a number of study cases in WP4 to validate its research outcomes. Within the project, there is a specific task that runs for the project duration that both informs and is informed by the activities in the project, including experimentation. This is shown in Figure 2, where WP4 on experimentation feeds into Task 1.1, which is intended to consolidate project results, and subsequently inform future activities. This process will iterate throughout the project and a series of deliverables emerging from WP1 (and specifically Task 1.1) that reflect on project learning will be produced. It can be seen that via this structure, there is a close coupling between the experimentation and research tasks in the project; the research tasks drive the experimentation process, which in turns feeds its results to back to the research work.

The four study cases that will be used in ResumeNet to validate our research output are listed below:

1. Resilient routing and medium sharing in Wireless Mesh Networks
   This scenario will investigate how to mitigate selfish routing behavior in wireless mesh networks. Furthermore, it will consider how to detect jellyfish attacks (where intermediate hops in multi-hop network manipulate packets (e.g., introduce additional delay) to reduce throughput. As an example, mechanisms (or a framework) for monitoring

and detection will be used to determine the existence of such selfish behavior (perhaps using information from multiple layers of the protocol stack and different topological locations), so that appropriate remediation can be undertaken.

2. Resilient forwarding in opportunistic networks
In this experiment, we intend to investigate aspects related to congestion management in opportunistic networks. That is dissemination strategies, resource management (e.g., data ageing), and resilience to attacks. The goal is to investigate the influence of different strategies on system behavior and performance to finally improve resilience. We will use the Haggle architecture for implementing the different strategies. Various aspects of the work carried out in WP1 and WP2 will be evaluated using this test case. For example, in WP2 we propose to develop a framework that can be used to adaptively remediate based upon the existence of a particular challenge and the system context. This framework could be evaluated in the context of this scenario, where node mobility, for example, is used to influence the different strategies to ensure data delivery (e.g., the network- and transport-layer protocols used).

3. Service-level resilience evaluation
In this study case we will investigate the use of peer-to-peer and virtualization technologies for resilience, which will be investigated in WP3 of the project. Hybrid peer-to-peer approaches will be evaluated for resilience, where essential services (e.g., in a VoIP environment) will be made unavailable to determine if service can be continued. With regard to the use of virtualisation techniques for resilience, we will examine the requirements and benefit of dynamic service mobility, taking both cold migration and live migration into account as a remediation mechanism. Restrictions on mobility, both on the service side and on the virtualization side, have to be identified. Additionally, appropriate control mechanisms to trigger a service migration will be defined.

4. Resilient smart environments
This case study will look at aspects of resilience in relation to smart environments, such as those to be found in the future home. Today's commercial offers for home terminals are diverse: multimedia cell phones (integrated in the domestic network through WiFi); digital TV decoder (VoD, visiophony) using Ethernet, WiFi or PLC, PCs (media center, VoIP); residential gateways holding a privileged position for shared services (e.g., data storage, peripheral equipments, distributed services); nomadic terminals (pocket computers, audio/video walkman). We will investigate the different challenges that can occur in this multi-device environment, using research that was developed in WP1 of the project on understanding challenges, and develop mechanisms to mitigate them.

For these test scenarios, tools will be developed that can emulate challenges to the normal operation of the systems that exist in the scenario. Furthermore, appropriate testbeds will be developed or extended for use in the project, as in the case of the ETHZ TIK testbed, discussed below. The actual experimentation facilities to be used in case studies will vary from in-house testbeds to those that are physically distributed, which may come from other EU R&D activities. Examples of the first case are the Wireless Mesh Network testbed of the ETHZ TIK group and the Haggle testbed maintained by the Uppsala University. These two testbeds are outlined in ResumeNet deliverable D4.1a. An example of the second alternative is the testbed that will come from the ANA project.

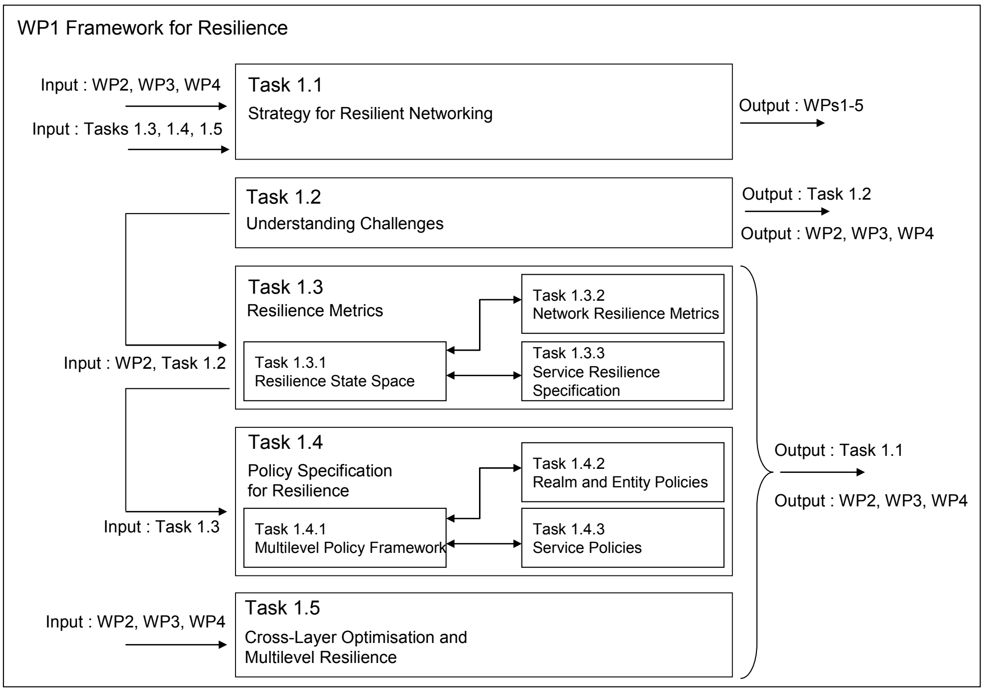

Figure 2: ResumeNet Work Package One Task Structure, showing the relationship of WP4 on experimentation with the fundamental research output of the project

2 Relation to testbeds so far

2.1 Existing Testbeds

The testbeds that will be used within the ResumeNet project have rather modest objectives, such that their proposed usage is to validate the project's research output, rather than be widely used by a larger community. To this end, we have a small integration overhead with other testbeds that are available. The exception to this is the study case that will evaluate service-level resilience. For this case, we propose to make use of PlanetLab Europe. We are open to make our experimental facilities available to other projects. The two existing testbeds we will use are (more detailed descriptions of these can be found in ResumeNet deliverable D4.1a):

2.1.1 The TIK-Testbed (TikNet)

The scenarios that consider wireless mesh networking will use the TIK-testbed, which is readily available at ETA Zurich, having already been used in prior research work. The testbed consists of twenty PCs that are connected both wired and tirelessly. The wired connectivity is used for configuring the testbed. The testbed is currently enhanced in two directions:

- with software modules that can emulate a number of possible node misbehaviors at the routing layer. The Click modular router software is used in this context.
- with software tools enabling monitoring of the state of nodes and links as well as the automatized configuration of the nodes' wireless interfaces.

2.1.2 The Haggle Testbed

The Haggle testbed allows to emulate a mobile opportunistic network and conduct repeatable tests in a controlled and easy to manage environment. The Xen virtual machine monitor is at the core of the testbed. Opportunistic network connectivity is emulated over a virtual ethernet bridge. Controlling connectivity with traffic filtering performs network topology changes. A graphical management console allows to start/stop nodes, control connectivity, and visualize the internal state of the diverse nodes and their interaction. The testbed is available as a software distribution that project partners can install on their own hardware. Federation would be possible through a virtual network but has not been

2.2 Testbeds to be developed

Within the context of the project, we intend to develop testbeds for our service-level resilience evaluation and smart home environments. These plans are described in ResumeNet deliverable D4.1a.

3 Criteria for the experimental facility

3.1 Access Policies

As mentioned earlier, the testbeds that will be used by the ResumeNet project will be in-house, and as such will have bespoke access policies. For this, we have sets of tools that allow local policies for access and reconfiguration to be undertaken. If policies were advocated as part of the FIREWorks coordination action, we would be willing to adjust our local testbeds in line with these.

3.2 Foreseen Challenges

One of the major challenges with all testbeds is sustainability – using testbeds incurs a continuing cost in terms of management and hardware renewal, for example. Beyond the end of project funding, because of these costs it is difficult to keep a testbed in a usable state.

3.3 Preferred Features

Below is a set of desired features for future testbeds:

- Within the context of the service-resilience use case in ResumeNet, there is need for conducting large-scale experimentation. For the moment, this has been done on PlanetLab, with up to 500 nodes being used. It would be desirable for this to be done on a much larger testbed, e.g., in the order of several thousand nodes.
- Another feature that would be desirable is flexible configuration of testbeds, for example, to deploy new operating system software on devices, or allow multiple configurations when appropriate, e.g., via virtualization. This should be achievable via a remote interface.
- Controlling the parameters of a testbed for experiments, e.g., the topology used, is often cumbersome and time-consuming. A desirable feature of future testbeds would be a user-friendly control interface (GUI) that would reduce the overhead of changing testbed parameters. Furthermore, such an interface could be used to monitor the state of the testbed during experiments.
- Reliability and availability of testbeds can be problematic. The notable example of this is PlanetLab, where nodes are often unavailable, and during peak periods (e.g., immediately prior to conference deadlines) can be heavily used. Ways to address this problem could include increased resources (therefore greater investment), and better access control policies so that resources can be shared between experiments to ensure more increased reliability (and therefore quality of results).
- A common issues with testbeds is that of ownership and access to resources. Transparent policies or simplification of access to resources that are owned by others would be desirable. PlanetLab, for example, has a model where access to resources is relatively straightforward because of the shared ownership of the testbed – participants contribute resources in return for access.