Plans for D7.7
The Security Report on the Final Project Release

Linda Cornwall, RAL

- Let them know what a lot we've done – even though we haven't solved everything!
- State which Security middleware tools have been developed within EDG.
- State which Security Middleware tools have been integrated with the various WP's software
- Tell them admin type stuff we've done – e.g. CA
- Indicate progress from the first testbed – I.e. more requirements satisfied!
- Say where work still needs to be done in future projects.

- Say we have a perfectly secure testbed (cos we haven't)

- Say we have satisfied all the requirements (which we haven't)

- Only describe what's turned on in the final testbed. (It's about the final project release – not just the testbed – and we have the tools to potentially do a lot more than the final testbed will do.)

- Not as simple as yes and no.
- E.g AUT-02 "Mutual Authentication between principals must be possible via use of certificates"
- Globus tools may satisfy (developed externally to edg)
- edg-java-security may satisfy (developed within edg)
- A WP may have more than 1 component, each of which may be integrated with 0 or 1 authentication method.

Suggested structure

- 1 Introduction (As EU spec)
- 2 Executive summary (As EU spec)
- 3 Externally developed Grid Security tools
  – A sentence or two to say what they are, and a ref.
  – Which requirements they address (and in which programming language or environment if applicable)
  – Which WPs use them.
  – Refer to the design and where they fit in
- 4 Grid security tools developed within the EDG project
  – As externally developed tools, but
  – Include even if not deployed or integrated with an other SW.
- 5 DataGrid Security Operations
  – Not only about tools – but a big admin effort too.
  – E.g.CA

- 6 Requirements discussion by area.
  - Table of Requirements for each area, and which tools can satisfy them
  - And whether the tools are fully integrated.
  - Discussion.
- 7 Is the Testbed secure?
  - Formal analysis? Can any be done?
  - Is the implementation secure?
- 8 Conclusion and future work.
  - Possibly include e.g. Browser Standard vs Proxy Service cert
  - Possibly include how combination of tools available could be used to e.g. satisfy WP10 req.
- Appendix – Possibly 1 requirement table per WP? Or 1 table for each language? E.g. 1 for java? Or repeat the line for each language if applicable? Or just 1 table in chapter 6?

To progress

- I could just get on with it – do everything as I think.
- Appropriate person to check I.e.
  - Those who developed a tool check I've got it right
    - Check e.g. references and which req are addressed.
  - Each WP check I've got their tool integration right and not missed anything.
    - Include checking which requirements are satisfied.
- Or - If you prefer – each WP could give me
  - A list of which tools they developed.
    - Preferred reference to the documentation about that tool.
    - Which requirements it addresses.
  - Which security tools are integrated into your S/W
    - Which requirements are satisfied.