

Sunday, April 27. 2008

Less known Solaris Features: Point-in-time copy with AVS

Okay, the next installment of the "Less known Solaris Features" series is online. This time i will discuss the feature Point in time copies with AVS. As i have to go through some theory at first, it's a quite long tutorial. It's the longest so far. Point-in-time copies are a rather unusual feature for small installations, but they are absolutely essential for many enterprise customers. And when you really think about it, there are many usecases even for small installations.

Part 1: Introduction

Part 2: Basics

Part 3: Independent copy

Part 4: Dependent copy

Part 5: Compact dependent copy

Part 6: Preparation of the test environment

Part 7: Starting a point-in-time copy

Part 8: Working with point-in-time copies

Part 9: Disaster Recovery

Part 10: Administration

Part 11: Conclusion

Have fun while trying out the feature!

Posted by Joerg Moellenkamp in English, Solaris at 11:04

Thanks once again for these great tutorials. Is AVS part of Solaris standard installation? Is there a separate license necessary?

I know this topic from storage vendors, e.g. EMC with timefinder products or Netapp with snapvault/snapmirror products. There you can do all of this stuff without presenting all devices to a host with storage internal copy mechanism. But you have to pay extra for this features. Copying is not done by the host in that case.

What are the main differences between AVS and storage based products?

Anonymous on Apr 27 2008, 17:03

AVS isn't part of the Solaris 10 standard installation. You have to buy an extra license. But AFAIK you can download the product and test it for 90 days.

You have to pay for it. It isn't cheap, but as far as i know it's much cheaper than the products from storage vendors.

The main advantage of doing this on the host, is the fact that you are independent from the hardware, for example, making a remote mirror from a expensive HDS to a cheap storage ... (or vice versa)

Anonymous on Apr 28 2008, 10:24