IP SAN

Dr Peadar Grant

March 14, 2024

1 iSCSI

IP SAN refers to SAN configurations that run over a standard IP network. [1] The most common type of IP SAN is iSCSI:

- iSCSI allows hosts to attach themselves to remote block devices called LUNs.[2]
  - Once attached, they are no different to a physical hard disk, partition or logical volume.
- iSCSI is very easy to get started using.
  - It works over standard IP networking and can obviously co-exist with other network usage.
  - Key software components are free and/or built-in to modern host OSes.

[1] We don't think about non-IP uses of Ethernet very often. Later on, we'll see how a different type of SAN works over ethernet without IP. Other specialised protocols exist too: many entertainment venues use the Dante protocol to provide low-latency audio connections over ethernet.

[2] LUNs are logical unit numbers, a carry-over from SCSI

1.1 Components

A **host** uses an **initiator** to attach to a specified **target**, . The target may present as one or more **LUNs**.

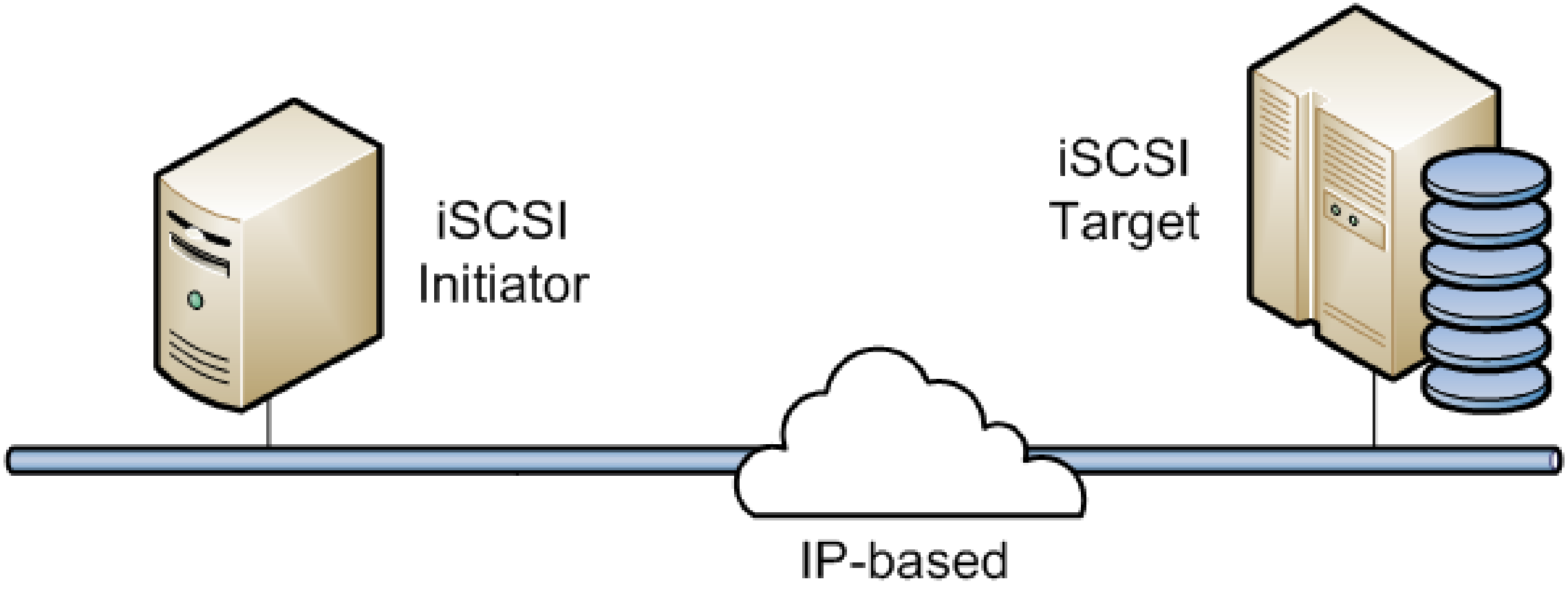

Figure 1: iSCSI initiator and target

1.2 iSCSI Initiator

- Once attached, the host is entirely responsible for creating / managing filesystems on the LUN.
  - *This is the key difference to Network Attached Storage (NAS).*
- Initiator can be either:

  Software-based: part of the host OS (usually kernel-level driver)

  Hardware-based: controller card or built-in to chipset on host:

  - Has *separate* IP configuration to host.
  - Usually separate physical port.
  - Can be *converged network adapter (CNA)* where both host networking and storage networking are on same physical port with different IP addresses.

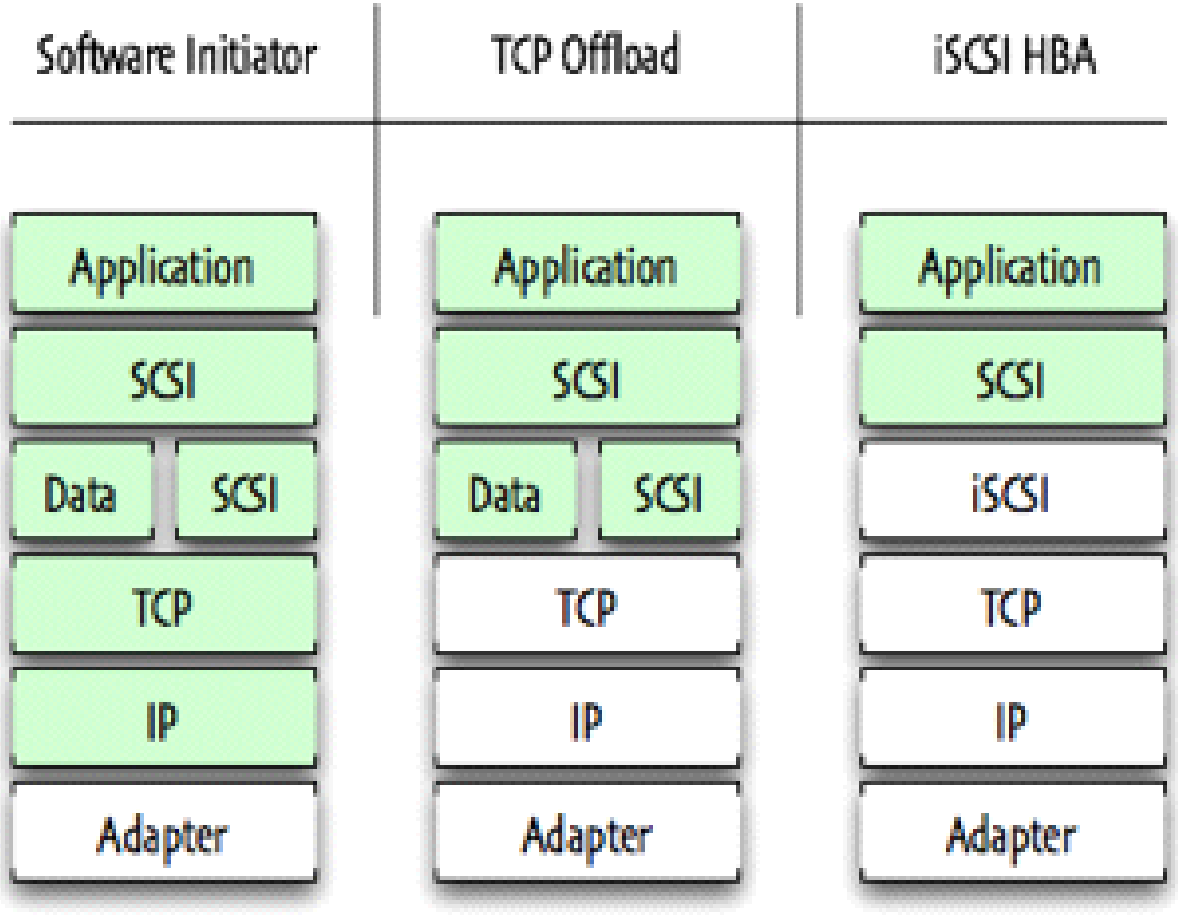

Figure 2: iSCSI offloading

1.3 iSCSI encapsulation

iSCSI carries SCSI commands over a standard TCP/IP network, Figure 3.

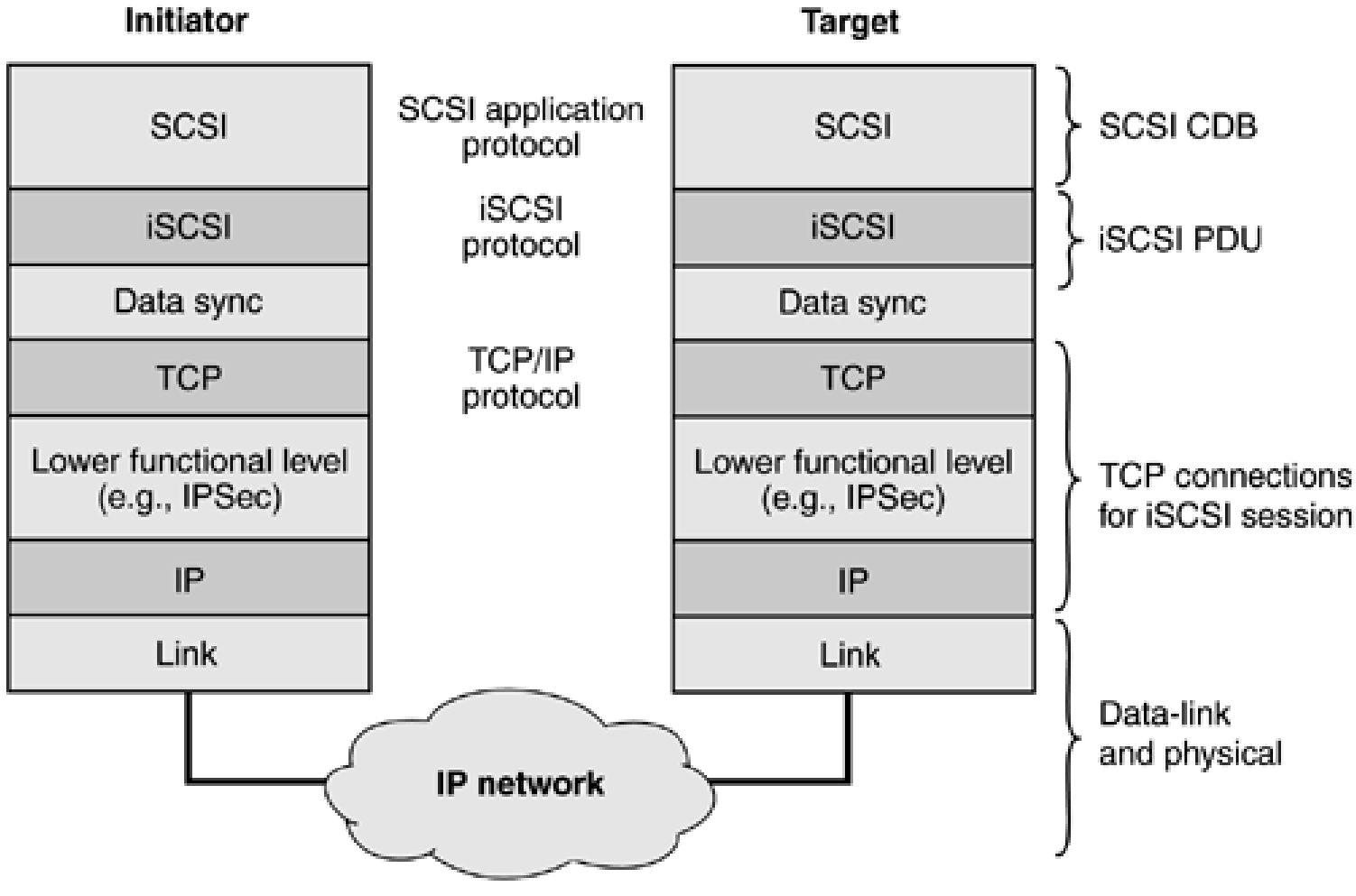

Figure 3: iSCSI stack

1.4 iSCSI Protocol data units (PDUs)

iSCSI encapsulates SCSI commands within iSCSI protocol data units (PDUs), Figure 4.

The encapsulation can be seen also in terms of how commands are processed, Figure 5.

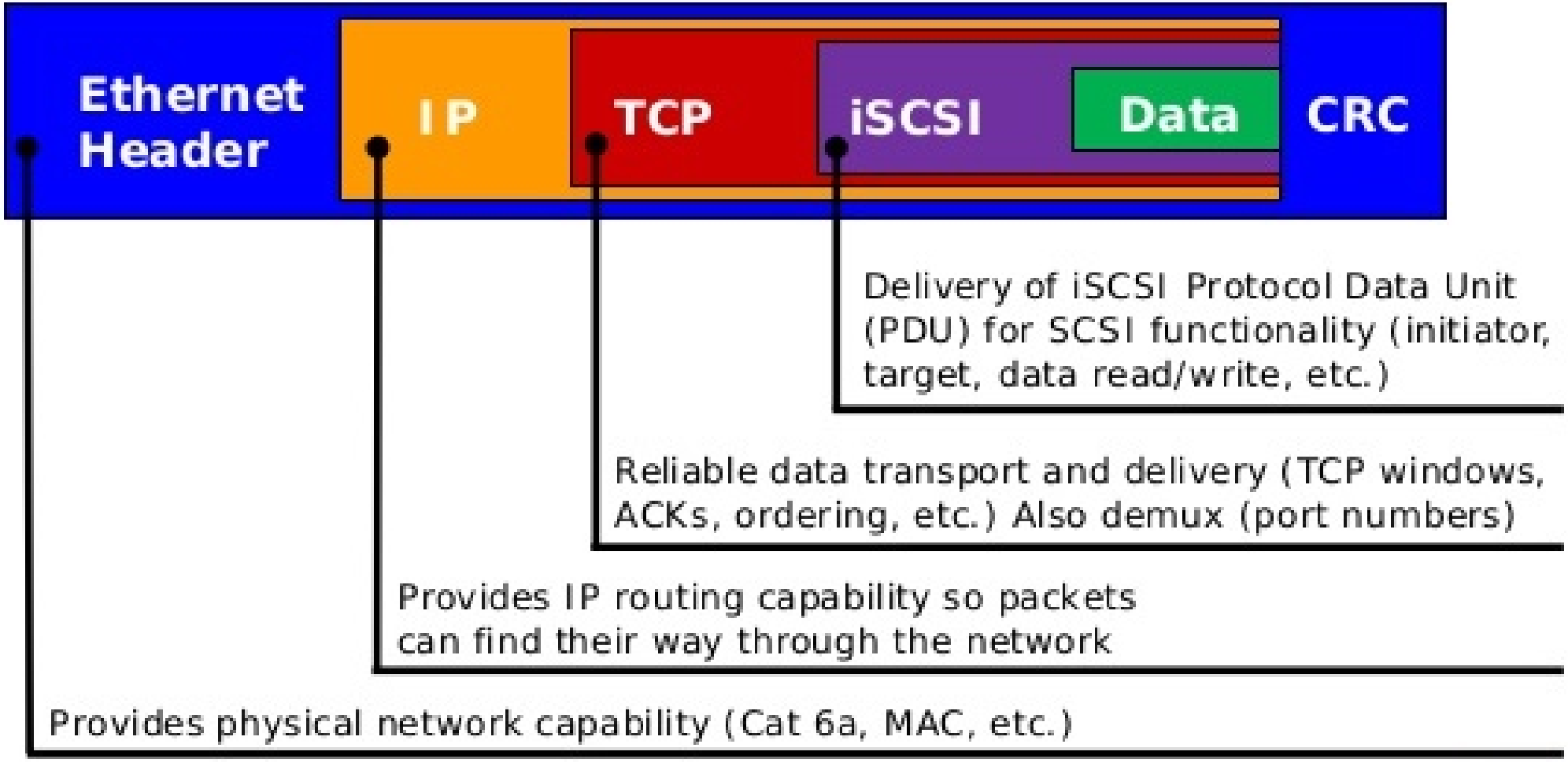

Figure 4: iSCSI encapsulation (EMC)

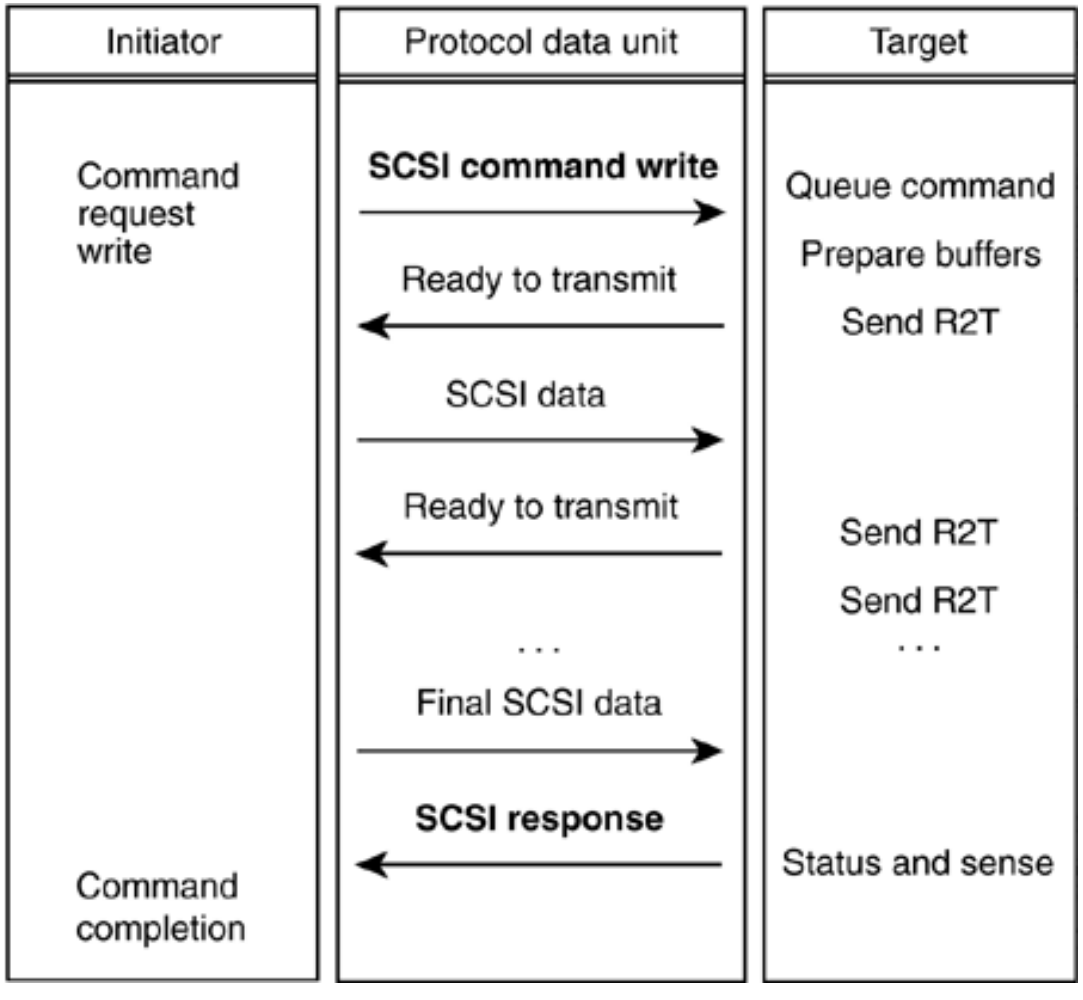

Figure 5: iSCSI write

2 iSCSI Targets

A **target** presents one or more **LUNs** to an initiator:

- Targets can be provided in a number of ways:
  - software target on a normal server OS
  - specialised SAN appliance
  - Some NAS units have iSCSI target capability also.
- **iSCSI initiators connect to targets that present one or more LUNs.**
- Remember that as a network, an iSCSI storage scheme could encompass multiple servers drawn from all of the above.

2.1 Dedicated storage appliances

- Often contain multiple drive holders.
- Can set up one or more RAID sets in multiple levels, possibly with hot spares.
- Can provision logical volumes on RAID sets (thin or thick provisioned)
- Presents as one or more targets, each with one or more LUNs.
- Tightly integrated hardware and custom OS (often Linux/Unix underneath).
- Designed for high availability, hot swap dual PSUs, redundant net interfaces.
- Often has management features like lights-out management card, SNMP support, web and CLI management.

2.2 General-purpose servers

- iSCSI target daemon can have one or more targets with one or more LUNS set up.
- LUNS can point to:

  Physical partition or any other block device

  Logical volume (since it's just a block device)

  Disk image file in a number of possible formats

2.3 NAS units

Primarily designed for file sharing often offer some iSCSI capabilities:

- Older units tended to use disk image files. More recent units tend to use LVM-like setup with iSCSI LUN provisioned as a logical volume.
- Cheap and easy way to experiment with iSCSI before committing to financial outlay.

2.4 iSCSI Connection

- To connect to a LUN we need to know: The IP address or DNS name of the machine / device providing the target service.
- The name of the target that is presenting the LUN we want.
- Can query list of available targets from server.
- The connection is made at the target level. Once made, all LUNs are visible to the initiator as block devices.

2.5 Target naming (IQNs)

iSCSI targets are named in a systematic way that can seem confusing at first. Each target has an iSCSI Qualified Name, or IQN. IQNs have 4 parts:

iqn literal string to denote this as an IQN.

naming authority usually the reversed domain name of the company (sometimes storage appliance vendor)

date when the naming authority assumed the name

target name string defined by the naming authority (the format of these vary across different target server systems)

An example of an IQN might be:

- `iqn.2019-12.ie.dkit.staffstorage` is an IQN for a target `staffstorage` by the `dkit.ie` naming authority who assumed control on `2019-12`.

2.6 Naming authority

In practice, the naming authority has nothing to do with DNS names other than by convention:

- For simple applications, once you know the DNS name / IP of the server and the name of the target, you can normally discover the IQN automatically.

Put simply, just accept that IQNs are strange-looking.

As long as you distinguish one target from another you will not have difficulty.

3 Authentication

iSCSI provides Challenge-Handshake Authentication Protocol (CHAP) for basic username / password authentication:

- CHAP secures a target rather than individual LUNs.
- CHAP is not particularly secure.
- Still on smaller networks, and to prevent mistakes, it's useful to require a username/password to connect to a target.

Storage appliances and software target servers normally have the CHAP authentication as an option that can be enabled on a particular target.

3.1 Mutual authentication

You can also set up the initiator to require a username/password from the target, known as mutual authentication.

Having a server authenticate to a client can seem odd, but it makes sure you've got the right disk connected and that sensitive data isn't written to an incorrect location.

4 LUN masking

LUN masking is where a target presents only a subset of its LUNs depending on what initiator is connecting.

The LUNs that are masked from a particular initiator do not appear to exist at all.

Support for LUN masking varies amongst target servers and storage appliances.

5 iSCSI initiator operations

We will walk through the process of connecting to a LUN from the point-of-view of the initiator.

We will not worry about how the LUN is provisioned at this point.

Scenario:

- A target is provisioned on a storage server with IP address `192.168.1.4`.
- Naming authority is `storage.com` starting from December 2019 (`2019-12`).
- Target is named `stuff`.
- There are two LUNs in this target.

5.1 Linux software initiator

There are two parts to iSCSI on Linux:

1. the kernel driver itself, and
2. the surrounding software to administer iSCSI connections
    - Most linux distributions nowadays use the open-iscsi package

Other UNIX variants like FreeBSD have similar functionality.

1. Host installs the `open-iscsi` package which provides the `iscsiadm` command. All `iscsiadm` commands need to be run as root (using sudo, omitted from steps below).
2. Host runs **discovery** on the server to see available targets:

   ```
   iscsiadm --mode discovery --type sendtargets --portal 192.168.1.4
   ```

   Note you can abbreviate mode, type and portal switches as `m, t, p` as per the man page. The available targets will appear (this has just one):

   ```
   192.168.1.4:3260,1 iqn.2019-12.com.storage:stuff
   ```

   May see a second line for each target if the server is running both IPv4 and IPv6.

3. Host **logs in** to the target:

   ```
   iscsiadm -m node -T iqn.2019-12.com.storage:stuff -p 192.168.1.4 --login
   # note here we use the short m, p switches
   ```

   Assuming the login step is successful, the LUNs on the target will appear as locally attached SCSI disks. So they will appear in `/dev` directory as `/dev/sd?` where ? is the drive letter.

4. The host then can partition and format these disks as if they were locally attached physical devices. Usually GPT/MBR single partition.

5. Host can disconnect by **logging out**:

   ```
   iscsiadm -m node -T iqn.2019-12.com.storage:stuff -p 192.168.1.4 --logout
   ```

   This is like plugging out a disk:

   - Therefore the filesystems need to be unmounted (ejected) before issuing the logout command.

6. Normally we would enable automatic login and ensure the iscsi service is set to start at boot-up.

5.2 Windows software initiator

Windows has a software initiator for iSCSI built-in to the operating system:

- This can be accessed graphically or via PowerShell.
- Once attached to a target, the LUN(s) appear in the Logical Disk Management display and can be partitioned and formatted etc as normal.

5.3 Hardware HBA

Hardware HBA allows attachment to iSCSI target without the knowledge of the host OS.

- Hardware HBA has its own independent connection to the network, separate to the host OS. It is a separate interface, with separate IP address etc.
- HBA normally appears to the host OS as a SCSI disk controller card.
- Some HBAs have a side-channel to the host OS (often appearing as a serial port) for management purposes. Some also include configuration tools / software.