DNS-01 certificates for package hosts

- `_acme-challenge.FQDN` is a CNAME pointing to `FQDN._acme.netdef.org`
  - FQDN is, as implied, the FULL host name in both cases.
    - `_acme-challenge.pkg.netdef.org. IN CNAME pkg.netdef.org._acme.netdef.org.`
  - it is always `_acme.netdef.org` - it does not matter if FQDN is under `netdef.org` or not.
    - `_acme-challenge.pkg.frrouting.org. IN CNAME pkg.frrouting.org._acme.netdef.org.`
- `_acme.netdef.org` is served by `ns-ch.netdef.org` (ONLY that server, there is no secondary, it makes no sense to have a secondary)
- DDNS updates with a TSIG key `certbot-key.` are enabled on that zone
- certbot is configured to use the `python3-certbot-dns-rfc2136` module to put the challenges into DNS using that TSIG key
- `certbot certonly --deploy-hook /etc/letsencrypt/renewal-hooks/deploy/01_deploy_pkg_servers.sh --dns-rfc2136 --dns-rfc2136-credentials /etc/letsencrypt/tsig.conf --agree-tos --manual-public-ip-logging-ok -d deb-us.netdef.org -d pkg-us.netdef.org -d rpm-us.netdef.org`
- the deploy-hook uses SSH to copy the keys to the target system
  - there is a special key for this on `ns-ch` in `/etc/letsencrypt/ssh_push_id`
  - this key is in `authorized_keys` on the package servers with `command="/etc/letsencrypt/ssh_receive.sh"` restriction
  - `/etc/letsencrypt/ssh_receive.sh` saves the key and reloads nginx

NOTE: the version of `python3-certbot-dns-rfc2136` on ns-ch did not support CNAMEs and was manually patched and marked with `apt-mark hold` .