

Basic syntax checking can be done with the 'filter\_username' policy. From the FreeRADIUS documents:

```
#      Filter the username
#
# Force some sanity on User-Name. This helps to avoid issues
# issues where the back-end database is "forgiving" about
# what constitutes a user name.
```

Example usage:

```
authorize {
    filter_username
    loop_prevent
    operator-name
    suffix
}
```

To filter out unwanted realms, such as 'hotmail.com' can be done simply by using the 'realm' command:

```
## Filter out NULL realms e.g. Username = fred

realm NULL {
}

## Filter out realms that aren't every going to be valid Govroam realms e.g.
Username = fred@hotmail.com

realm "~hotmail\\.com$" {
}

realm "~hotmail\\.co\\.uk$" {
}

realm "~.*\\.3gppnetworks\\.org$" {
}

realm "~.*\\.3gppnetwork\\.org$" {
}

realm "~gmail\\.com" {
}

realm "~googlemail\\.com" {
}

realm "~live\\.com" {
}

realm "~outlook\\.com" {
```

```
}  
  
realm "~yahoo\\.com" {  
}  
  
realm "~yahoo\\.cn" {  
}  
  
realm "~unimail\\.com" {  
}  
  
realm "~yahoo\\.co\\.uk" {  
}  
  
realm "~myabc\\.com" {  
}
```

alternatively

```
check_bad_realms {  
    if (&User-Name) {  
        ## reject usernames that end with realms that are never  
going to be govroam/eduroam realms  
        ## is there a way to automate this??!  
  
        ## stuff here will be matched for just eduroam  
        if (&control:Proxy-To-Realm =~ /^PROXY-JANET-EDUROAM$/) {  
  
            if (&User-Name =~ /test\\.net$/) {  
                update request {  
                    &Module-Failure-Message +=  
'Rejected: Bad Realm'  
                }  
                reject  
            }  
  
            if (&User-Name =~ /@.*bathnes/) {  
                update request {  
                    &Module-Failure-Message +=  
'Rejected: Bad Realm'  
                }  
                reject  
            }  
        }  
    }  
  
    ## stuff here will be matched for just govroam  
    if (&control:Proxy-To-Realm =~ /^PROXY-JANET-GOVROAM$/) {  
  
        if (&User-Name =~ /that\\.host$/) {
```

```
                update request {
                    &Module-Failure-Message +=
'Rejected: Bad Realm'
                }
            }
        }

## stuff here will be matched for both govroam and eduroam
## Common realm provided by phone manufacturers by default
if (&User-Name =~ /3gppnetwork\.org$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

## Username is not an email address.
if (&User-Name =~ /gmail\.com$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

## Ends in '.'
if (&User-Name =~ /bath\.ac\.uk\.+$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

## Ends in a single character - all TLDs are 2+ characters.
if (&User-Name =~ /\.[a-z0-9]$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

## All subrealms should have more than one character
if (&User-Name =~ /\.[a-z0-9]\.uk$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
}
```

```
        reject
    }

    ## Misspellings and misunderstandings
    if (&User-Name =~ /ac\.uk\.com$/) {
        update request {
            &Module-Failure-Message += 'Rejected: Bad
Realm'
        }
        reject
    }

    if (&User-Name =~ /\.acuk$/) {
        update request {
            &Module-Failure-Message += 'Rejected: Bad
Realm'
        }
        reject
    }

    if (&User-Name =~ /@ac\.uk$/) {
        update request {
            &Module-Failure-Message += 'Rejected: Bad
Realm'
        }
        reject
    }

    ## Starts with a '.'
    if (&User-Name =~ /\.bath\.ac\.uk$/) {
        update request {
            &Module-Failure-Message += 'Rejected: Bad
Realm'
        }
        reject
    }

    ## Some typical inappropriate realms.
    if (&User-Name =~ /@myabc\.com$/) {
        update request {
            &Module-Failure-Message += 'Rejected: Bad
Realm'
        }
        reject
    }

    if (&User-Name =~ /@.*hotmail\.co/) {
        update request {
            &Module-Failure-Message += 'Rejected: Bad
Realm'
```

```
    }
    reject
}

if (&User-Name =~ /@.*live\.co$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

if (&User-Name =~ /\\.local$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

if (&User-Name =~ /\\.guest$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

if (&User-Name =~ /@.*gmail\.co/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

if (&User-Name =~ /googlemail\.com$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

if (&User-Name =~ /outlook\.com$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}
}
```

```
if (&User-Name =~ /@.*yahoo\.c/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

if (&User-Name =~ /unimail\.com$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

if (&User-Name =~ /blueyonder\.co\.uk$/) {
    update request {
        &Module-Failure-Message += 'Rejected: Bad
Realm'
    }
    reject
}

} else {
    update request {
        &Module-Failure-Message += 'Rejected: erm... WTF?!'
    }
    reject
}
}
```

Many thanks to Matt Richards from Bath University for this code.

From: <https://wiki.govroam.uk/dokuwiki/> - **Govroam**

Permanent link: [https://wiki.govroam.uk/dokuwiki/doku.php?id=siteadmin:realm\\_filtering](https://wiki.govroam.uk/dokuwiki/doku.php?id=siteadmin:realm_filtering)

Last update: **2019/12/05 10:51**

