

Firebase Database Rules

Mobile Computing - Android

Objectives

- Students will be able to:
 - Identify common rule applications
 - Create custom rules

Introduction

- Firebase has 4 basic kinds of rules

.read

Describes if and when data is allowed to be read by users.

.write

Describes if and when data is allowed to be written.

.validate

Defines what a correctly formatted value will look like, whether it has child attributes, and the data type.

.indexOn

Specifies a child to index to support ordering and querying.

Rules on path

```
{  
  "rules": {  
    "foo": {  
      ".read": true,  
      ".write": false  
    }  
  }  
}
```

For the path /foo and deeper everyone can read, no-one can write.

Unlike what you would expect, shallow rules override deeper rules.

Rules using variables

```
{
  "rules": {
    "users": {
      "$uid": {
        ".write": "$uid === auth.uid"
      }
    }
  }
}
```

For the path `/users/anID/`. `Auth.uid` will be null before authentication and will have the `userid` after. If the `userid` for the data matches then we grant write permission.

`$uid` is a dynamic variable that will refer to a particular user.

Path Capture

```
{
  "rules": {
    "rooms": {
      // this rule applies to any child of /rooms/, the key for each room id
      // is stored inside $room_id variable for reference
      "$room_id": {
        "topic": {
          // the room's topic can be changed if the room id has "public" in it
          ".write": "$room_id.contains('public')"
        }
      }
    }
  }
}
```

For the path /rooms/anID/topic. We capture the room_id and allow topic to change if the room id contains public.

Path Validations

```
{
  "rules": {
    "widget": {
      // a widget can have a title or color attribute
      "title": { ".validate": true },
      "color": { ".validate": true },

      // but no other child paths are allowed
      // in this case, $other means any key excluding "title" and "color"
      "$other": { ".validate": false }
    }
  }
}
```

For the path /widget/title and /widget/color. We validate on title and color. Any other path is excluded

Simple Rule for Example

```
{  
  "rules": {  
    ".read": true,  
    ".write": "auth !== null"  
  }  
}
```

If we want to specialize access based on the user id, we should have paths based on the uid.

Reference

- <https://firebase.google.com/docs/database/security/rules-conditions>

Questions