

Internal Function - Unit Testing

`uint64_t get_nft_number_from_name(name userid); -- version 10a (2023-09-30)`

```
// get_nft_number_from_name(name userid) -- version 10a (2023-09-30)
// a-z & 1-5 are valid sName 1=1, 5=5, a=6, y=30, z=0
uint64_t nftscribe::get_nft_number_from_name(name userid) {
    vector<string> aName = split(userid.to_string(), ".");
    check(aName.size() == 2, "Invalid userid specified for user.");

    string sName = aName[0]; //max 5 digits
    string sSuffix = aName[1]; //max 6 digits

    check((sName.size() <= 5) && (sName.size() >= 1), "get_nft_number_from_name - Username of userid is too long / too short, between 5 and 1 digits inclusive. ");
    check((sSuffix.size() <= 6) && (sSuffix.size() >= 1), "get_nft_number_from_name - Suffix of userid is too long / too short, between 6 and 1 digits inclusive. ");

    // Helper function to convert character to its corresponding value
    auto char_to_value = [](char c) -> uint64_t {
        if (c >= '1' && c <= '5') return c - '0';
        if (c >= 'a' && c <= 'y') return c - 'a' + 6;
        if (c == 'z') return 0; // Special case for 'z'
        return 0; // Default, shouldn't happen with checks in place
    };

    uint64_t result = 0;
    for (size_t i = 0; i < sName.size(); ++i) {
        result *= 31; // Base 31
        result += char_to_value(sName[i]);
    }

    return result;
}
```

Testing the z.suffix to yyyyy.suffix breadth of base 31 NFT # possibilities.

PASSED Max Test (yyyyy.nft)

```
uint64_t name_test = get_nft_number_from_name(name("yyyyy.nft"));
check(false, "name_test=" + to_string(name_test));
```

Result Matches $(31^5) - 1$

```
assertion failure with message:
name_test=28629150
```

```
"code" 500
"message" "Internal Service Error"
```

PASSED Min Test (z.nft)

```
uint64_t name_test = get_nft_number_from_name(name("z.nft"));
check(false, "name_test=" + to_string(name_test));
```

Result Matches 0

```
assertion failure with message:
name_test=0
```

```
"code" 500
"message" "Internal Service Error"
"error"
```

PASSED Threshold Test (yy.nft) to (1zz.nft) Requires no overlap of value

```
uint64_t name_test = get_nft_number_from_name(name("yy.nft"));
check(false, "name_test=" + to_string(name_test));

post_index_posts( self, network_id,value);
```

```
assertion failure with message:
name_test=960
```

```
"code" 500
"message" "Internal Service Error"
```

```
uint64_t name_test = get_nft_number_from_name(name("1zz.nft"));
check(false, "name_test=" + to_string(name_test));

post_index_posts( self, network_id,value);
```

```
assertion failure with message:
name_test=961
```

Result was correctly increased by 1 in value.

PASSED Boundary Tests - All Should Fail (.nft), (yyyyyy.nft), (aa.nft.vi), (test)

(.nft) - correct

```
uint64_t name_test = get_nft_number_from_name(name(".nft"));
check(false, "name_test=" + to_string(name_test));
```

▼ Result Error

```
assertion failure with message:
get_nft_number_from_name - Username of
userid is too long / too short, between 5
and 1 digits inclusive.

"code" 500
"message" "Internal Service Error"
"error"
  "code" 3050003
  "name" "eosio_assert_message_exception"
  "what" "eosio_assert_message assertion fail
ure"
```

(yyyyyy.nft) - correct

```
uint64_t name_test = get_nft_number_from_name(name("yyyyyy.nft"));
check(false, "name_test=" + to_string(name_test));
```

```
assertion failure with message:
get_nft_number_from_name - Username of
userid is too long / too short, between 5
and 1 digits inclusive.
```

```
"code" 500
```

(aa.nft.vi) - correct

```
uint64_t name_test = get_nft_number_from_name(name("aa.nft.vi"));
check(false, "name_test=" + to_string(name_test));
```

▷ Ricardian

▼ Result

```
assertion failure with message: Invalid
userid specified for user.
```

```
"code" 500
```

```
"message" "Internal Service Error"
```

```
"error"
```

```
"code" 3050003
```

```
"name" "eosio_assert_message_exception"
```

(test) - correct

```
uint64_t name_test = get_nft_number_from_name(name("test"));
check(false, "name_test=" + to_string(name_test));
```

▼ Result

```
assertion failure with message: Invalid
userid specified for user.
```

```
"code" 500
```