

Rnssp Part 1

2:30PM EST | November 17th, 2021 | 2021 SyS Symposium

Trainer: Roseric Azondekon, PhD, MPH, MS – CDC/DDPHSS/CSELS/DHIS/SDB



Council of State and Territorial Epidemiologists



Rnssp: A Signature R package for the National Syndromic Surveillance Program (NSSP)

G. Roseric Azondekon, PhD, MPH, MS

Health Scientist

Data Science and Innovation Team

Surveillance and Data Branch

Division of Health Informatics and Surveillance

2021 Syndromic Surveillance Symposium

November 18, 2021



Outline



- Learning Objectives
- Purpose
- Design Approach
- Secured Credentials Setup and Data Pull Interface
- Vignettes and Templates
- Continuous Integration/Continuous Deploy
- Demonstration

Learning Objectives

At the end of this training, you will be able to:

1. Understand the Rnssp package design philosophy
2. Install the Rnssp package in your R environment
3. Pull application programming interface (API) data using the Rnssp data pull interface
4. Apply anomaly detection and trend classification algorithms via the Rnssp package
5. Expand the Rnssp installation by adding NSSP report templates
6. Navigate the Rnssp online documentation

Why an R Package for NSSP?

To improve reusability, reproducibility, and distribution of the analytical resources developed at NSSP, specifically, we intend to:

- Support technical capacity building with our Community of Practice (CoP)
- Support the broad range of technical ability in the CoP
- Guide new users with multiple vignettes
- Provide better options for managing user credentials
- Secure access to NSSP-ESSENCE APIs
- Use ESSENCE APIs to support greater interoperability with other apps
- Streamline analytic tools
- Improve collaboration by sharing report templates, scripts, and other NSSP resources

Focus Areas

- Provide better options for managing user credentials
- Secure access to NSSP-ESSENCE APIs

Design Approach

Traditional Object-Oriented Programming (OOP) Approach in R implemented via the R6 OOP system, an implementation of encapsulated OOP for R.

OOP Core Principles:

- Abstraction
- Encapsulation
- Polymorphism
- Inheritance

Secured Credentials Setup and Data Pull Interface

Credentials class:

- `$new(username, password)` to create a new NSSP user profile
- `$get_api_response(url)` to get API response from ESSENCE
- `$get_api_data(url, fromCSV = FALSE, ...)` to pull API data from ESSENCE
- `$get_api_tsgraph(url)` to specifically pull ESSENCE Time Series graph and API response

NSSPContainer class to store encrypted information in a container object.

Secured Credentials Setup and Data Pull Interface

Setting NSSP Credentials

The problem with the old **Keyring** method:

```
library(keyring)
key_set(service = "essence", username = "<username>")
# These two lines print your Credentials in plain text
key_list("essence") # prints your username(s) in plain text
key_get("essence", key_list("essence")[1,2]) # prints your PASSWORD in plain text
```

The **Rnssp** way:

```
library(Rnssp)
# Rnssp encrypts your credentials and hides it into the 'myProfile' object
myProfile <- Credentials$new(
  username = askme("Enter your username"),
  password = askme()
)
```

Secured Credentials Setup and Data Pull Interface

Setting NSSP Credentials

```
myProfile # or str(myProfile)
```

```
<NSSPCredentials>
```

```
Public:
```

```
clone: function (deep = FALSE)  
get_api_data: function (url, fromCSV = FALSE, ...)  
get_api_response: function (url)  
get_api_tsgraph: function (url)  
initialize: function (username, password)
```

```
Private:
```

```
..__: NSSPContainer, R6  
..password: NSSPContainer, R6  
..username: NSSPContainer, R6
```

Secured Credentials Setup and Data Pull Interface

Setting NSSP Credentials

Keyring, httr, jsonlite:

```
# URL from JSON ESSENCE Alert List Detection Table API
url <- paste0("https://essence.syndromicsurveillance.org/nssp_essence/api/",
             "alerts/regionSyndromeAlerts?end_date=31Jan2021&start_date=29Jan2021")
# Data Pull from ESSENCE
api_response <- httr::GET(url,
                        httr::authenticate(key_list("essence")[1, 2],
                                           key_get("essence",
                                                  key_list("essence")[1, 2])))
api_response_json <- httr::content(api_response, as = "text")
api_data <- jsonlite::fromJSON(api_response_json)
```

With Rnssp:

```
# URL from JSON ESSENCE Alert List Detection Table API
url <- paste0("https://essence.syndromicsurveillance.org/nssp_essence/api/",
             "alerts/regionSyndromeAlerts?end_date=31Jan2021&start_date=29Jan2021")
# Data Pull from ESSENCE
api_data <- myProfile$get_api_data(url)
```

Focus Area

Streamline analytic tools

Exported Functions and Classes

`add_rmd_template()`

Utility to Add a New RMarkdown Template

`alert_ewma()`

Exponentially Weighted Moving Average (EWMA)

`alert_mar()`

Multiple Adaptive Regression

`askme()`

Password Prompt Utility

`change_dates()`

Utility to Change the Start and/Or End Dates in an ESSENCE API URL

`classify_trend()`

Trend Classification for Proportions/Percentages

`clean_text()`

Removes stopwords and common patterns of Chief Complaints

`Credentials`

A `NSSPCredentials` Class Representing a Credentials object

`list_templates()`

List Available Report Templates

`NSSPContainer`

A `NSSPContainer` Class to store a value in a container object

`remove_rmd_template()`

Utility to Remove an Existing RMarkdown Template

`Rnssp`

Tools, Functions, Shapefiles data, and Rmarkdown templates to work with Syndromic Surveillance data via NSSP ESSENCE

`Rnssp_vignettes()`

Browse Rnssp vignettes

Available Datasets

```
data(package = "Rnssp")
```

Data sets in package 'Rnssp':

cbsa_sf	US Core-Based Statistical Area Shapefile
county_sf	US Counties Shapefile
hhs_sf	US Health and Human Services Regions Shapefile
state_sf	US States Shapefile
territories_sf	US Territories Shapefile
tribal_sf	US Native American Lands Shapefile

```
?state_sf # View documentation for state_sf dataset
```

Focus Area

Improve collaboration by sharing report templates, scripts, and other NSSP resources

Report Templates



<https://github.com/CDCgov/Rnssp-rmd-templates>

```
# List available templates
```

```
list_templates()
```

```
# List available templates info as table
```

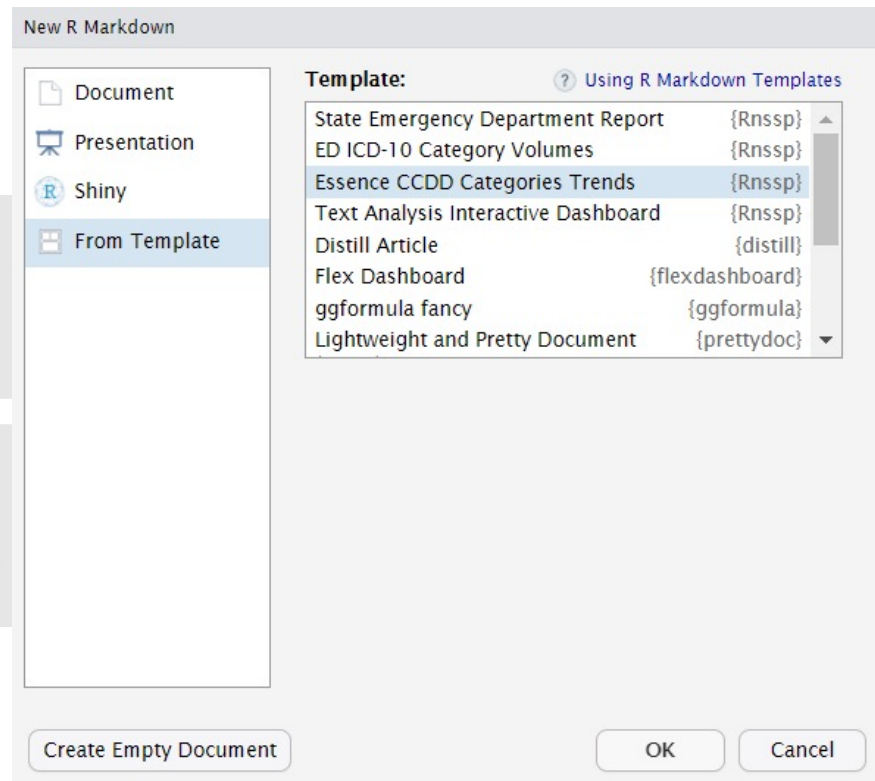
```
list_templates(as.table = TRUE) %>% View()
```

```
# Adds `text_mining` template to current Rnssp installation
```

```
add_rmd_template(template_name = "text_mining")
```

```
# Remove `state_ed_report` template from current Rnssp installation
```

```
remove_rmd_template("state_ed_report")
```



Focus Area

Support technical capacity building with our Community of Practice

Online Documentation

Online Documentation available at: <https://cdc.gov.github.io/Rnssp>



Rnssp

Overview

The Rnssp R package is a catalog of data processing and analytics tools, templates, and functions commonly used across the National Syndromic and Surveillance Program at the Centers for Disease Control and Prevention (CDC). Its goal is to improve code reproducibility, standardize and document reusable functions, facilitate the sharing of routine reports across the NSSP Community of Practice (CoP), and better engage the NSSP CoP.

Installation

You can install the development version of Rnssp from GitHub:

```
# install the Rnssp package
devtools::install_github("cdcgov/Rnssp")
```

Usage

This is a basic example which shows you how to:

- use the Rnssp package to create an NSSP user profile
- and use the NSSP user profile to pull data from ESSENCE via the ESSENCE API

```
library(Rnssp)

## Creating an NSSP user profile
myProfile <- Credentials$new(
  username = askme("Enter your username: "),
  password = askme()
)

## Inspect your `myProfile` confirming that username and password are completely hidden
myProfile
```



Links

Browse source code at
<https://github.com/CDCgov/Rnssp/>

Download build release at
<https://github.com/CDCgov/Rnssp/releases/>

Report a bug at
<https://github.com/CDCgov/Rnssp/issues>

Explore all Rnssp vignettes at
<https://cdc.gov.github.io/Rnssp/articles>

License

Full license

Apache License (>= 2)

Community

Contributing guide

Code of conduct

Developers

Gbedegnong Roseric Azondekon

Author, maintainer

All authors...

Dev status

Rnssp Pipeline passing

repo status Active

lifecycle stable

DOI 10.5281/zenodo.5034130

CRAN not published

Rnssp_vignettes()

All vignettes

Introduction to the Rnssp package

Text Mining Example

Anomaly Detection and Trend Classification

Continuous Integration/Continuous Deploy

```
3  on:
4    push:
5      branches:
6        - master
7        - main
8
9  name: Rnssp Pipeline
10
11 jobs:
12   node-docker:
13     runs-on: ubuntu-latest
14     container:
15       image: rocker/geospatial:4.0.2-ubuntugis
16     steps:
17       - name: Checkout Rnssp repository code
18         uses: actions/checkout@v2
19
20       # Performs a clean installation of all dependencies
21       - name: Install Dependencies
22         run: |
23           R -e "remotes::install_deps(dependencies = TRUE)"
24
25       # Performs Rnssp package test
26       - name: Check
27         run: |
28           R -e 'devtools::test()'
29           R -e 'devtools::load_all()'
```

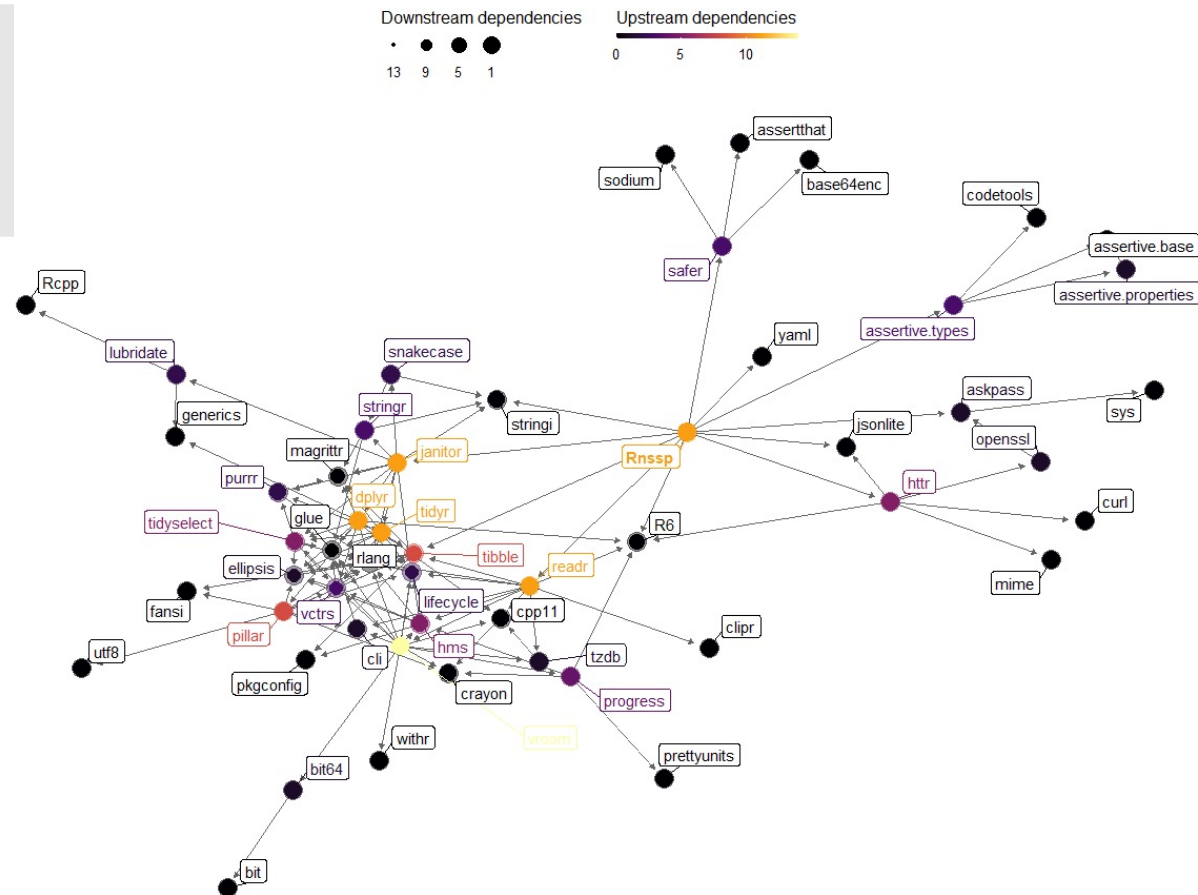
Package Dependency

```
# Rnssp package installation
devtools::install_github("cdcgov/Rnssp")

# Alternatively...
remotes::install_github("cdcgov/Rnssp")
```



<https://github.com/CDCgov/Rnssp>



Rnssp Pipeline **passing**
repo status **Active**
lifecycle **stable**
DOI **10.5281/zenodo.5034130**
CRAN **not published**

Demonstration



- ✓ Provides better options for managing user credentials
- ✓ Secures access to NSSP-ESSENCE APIs
- ✓ Streamlines analytic tools
- ✓ Improves collaboration by sharing report templates, scripts, and other NSSP resources
- ✓ Supports technical capacity building with our Community of Practice

Acknowledgments

Package Dev Contributors

- Roseric Azondekon [[POJ2](#)]
- Michael Sheppard [[OUL2](#)]

Template Dev Contributors

- Kelly Carey [[QLV5](#)]
- Michael Sheppard [[OUL2](#)]
- Roseric Azondekon [[POJ2](#)]
- Aaron Kite-Powell [[LYV8](#)]
- Sarah Chronister (Epidemiologist,
Maricopa County, AZ)

Supervision & Tech Support

- Aaron Kite-Powell [[LYV8](#)]
- Douglas Wirtz [[KSF4](#)]
- Roseric Azondekon [[POJ2](#)]

Questions?

gazondekon@cdc.gov



For more information, contact CDC
1-800-CDC-INFO (232-4636)
TTY: 1-888-232-6348 www.cdc.gov

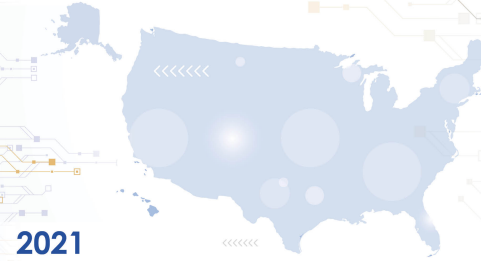
The findings and conclusions in this report are those of the authors and do not necessarily represent the official position of the Centers for Disease Control and Prevention.



National Syndromic Surveillance Program

Community of Practice

Resources that Advance the Science and
Practice of Syndromic Surveillance



2021

Syndromic Surveillance Symposium

Week of November 15, 2021



KAHUINA
CONSULTING