# TEVIN ETR – CoreEngine1 – Software Analysis

*© TEVIN INTERNATIONAL LIMITED – COPYRIGHT 2018 – all rights reserved*

## Protecting the script code

Python scripts as well as all other scripting languages like jscript or php can be encrypted and protected using <http://www.enscryption.com/faq.html#heading1lic>

This tool looks to be the ULTIMATE ANSWER not only on how to protect the code but also how to protect against adding unwanted software on the system that is running the code!

**Pricing: you pay 110 Euro** and can use it for 2 months to encrypt any scripts you have. The 2 months period refers to the time you have to use the tool. Once your scripts are encrypted they remain encrypted as long as you like.

This tool offers many more benefits apart from copy protection: you can stop execution of unwanted software, can control versions and have the script stop working after a certain time.

<https://www.scamadviser.com/check-website/enscryption.com>

**The enscryption site gets a TRUST RATING OF 90% and comes from USA, aged 17 years.**

## The general idea – execution OS and platform

The software is designed to be used as **the core engine** in all types and forms of RETAIL, RESTAURANT and ACCOUNTING applications. This means we intend to use the CoreEngine1 into **embedded ECR**, **Android devices**, **POS systems**, **Fiscal Printers** and **Server or Cloud** applications.

The software is written in PYTHON 3.6

The IDE used for development is the PYCHARM COMMUNITY EDITION

It is meant to run in UBUNTU 16.04 LTS as well as in EMBEDDED UBUNTU in NANO NEO SBC running Ubuntu Kernel 4.0 which is a low-power, NOT graphics intensive version offered for NANO devices.

## GitHub version control

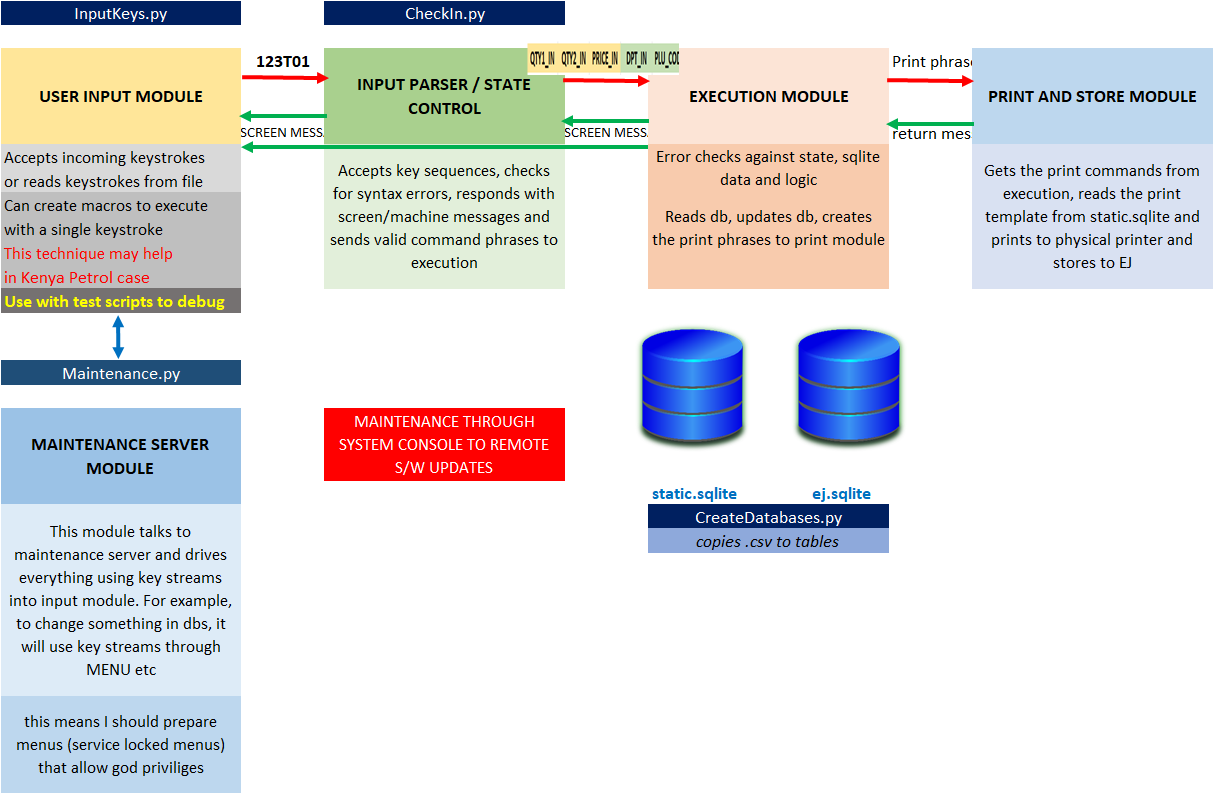
I have created a free (this means PUBLIC repository) account at <https://github.com>

## Block Diagram – Modules and databases

The engine can be at one of 3 states of operation: the SALES MODE STATE is the most complicated of the three as it executes all transactions coming in from any type of input. The REPORTS STATE and the PROGRAMMING state are not executing transactions but report or update the databases.

You CAN move from one state to the other EVEN IN THE MIDDLE of a transaction depending on RULES embedded in the various modules.

### SALES MODE STATE block diagram and source modules:



## Github: started public account to use Version Control in PyCharm

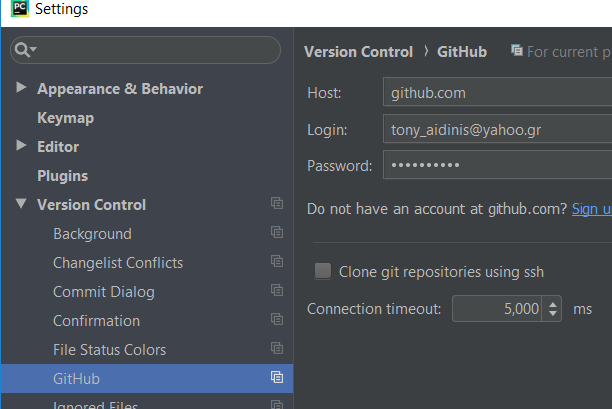
Account login: [tony\_aidinis@yahoo.gr](mailto:tony_aidinis@yahoo.gr) / passw: Quar2dife4

User name is **geta4tvn**

Created a repository as **geta4tvn/CoreEngineOne**

Public, initialize with a readme

Then, in pycharm I set up VCS (version control system) like this:



Can't finish GitHub sharing process

*Successfully created project 'CoreOne' on GitHub, but initial commit f ailed:*

*\*\*\* Please tell me who you are.*

*Run*

*git config --global user.email "you@example.com"*

*git config --global user.name "Your Name"*

*to set your account's default identity.*

*Omit --global to set the identity only in this repository.*

*fatal: unable to auto-detect email address (got 'taidi@WINDOWS-PH2ECTO.(none)') during executing git "C:\Program Files\Git\bin\git.exe" -c core.quotepath=false -c log.showSignature=false commit -m "Initial commit" --*

### To use VCS in pycharm you need to install git and setup git with the email and username at gith

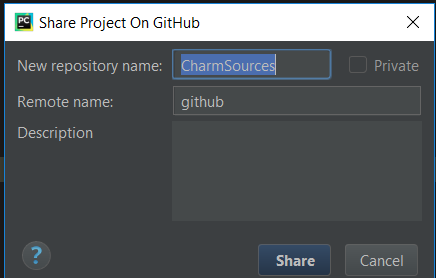
So, I install git, then run the following:

git config --global user.email "tony\_aidinis@yahoo.gr"

git config --global user.name "geta4tvn" (that’s the name I use in github account.

After this is setup, I do an initial commit through pycharm like this:

VCS > Import into version control > Share Project in GitHub



I give EngineOneCore and have the same name in “Remote name”

PRESS ? HERE TO GET TO HELP about how to

The github pops up asking me to login, and finally everything works fine and I get the message in the EVENT LOG panel: **Successfully shared project on GitHub: EngineOneCore**

This action is only when you CREATE the MASTER REPOSITORY.

To COMMIT or create BRANCHES what do you do?

### First you need to create a branch to work on – go VCS > Git > Branches

All source files and sqlite etc are into the MASTER repo EngineOneCore

Now I want to create a branch: VCS > Git > Branches and press **+** to add a new branch.

Result: **Branch EngineOneCore-Tony was created**

### How to commit changes to this branch? Go VCS > Commit

Going to VCS > Commit a “Commit Changes” window opens up that shows the DIFFERENCES and has a lot other tricks that I need to study…

### ALL FILES are Committed / versioned, not just the .py

Used sqlite studio to add a column to ejournal ej table so we can have a Type column for each receipt: 0 will be receipt, 1 will be Invoice, 2 will be Credit Note etc.

Then, I hit VCS > Commit and YES, the journal.sqlite was found that has changed so it would go into commit. The DIFFERENCES were NOT shown because the \*.sqlite isn’t a pycharm file so system doesn’t know which app to use to open it

# 1 October 2018

## Install git on TurboX

I don’t know if pycharm will work the same in DELL and in TURBOX, I mean use the same repository at github.

First, I installed git

Now, on which branch I’ll work in TurboX? The same I work with DELL?

The steps I took:

### Install git and configure it

After installing git in Turbox, I use the command panel to configure it:

git config --global user.email "tony\_aidinis@yahoo.gr"

git config --global user.name "geta4tvn" (that’s the name I use in github account.

(pointing to the same github account as with dell)

### Pycharm settings

Go to File > settings > git

There I need to point to the root folder where git is installed: This is C:/Program Files/Git/bin/git.exe

Then test this and I get an OK by pycharm

Then, go to File > settings > github

There, I enter my github login name and password: [tony\_aidinis@yahoo.gr](mailto:tony_aidinis@yahoo.gr) / Quar2dife4

### Error: no git repository found in folder

From pycharms’ event log:

10:58 πμ Invalid VCS root mapping

The directory D:\000 W…\OUR ETR\CharmSources is registered as a Git root, but no Git repositories were found there.

Configure...

I go configure but the folder there is the valid folder I want to use. So how do I create a git repository there?

You have to tell git to create a repository and add your project to that repository.

I found this guide:

<http://kbroman.org/github_tutorial/pages/init.html>

And I did this:

Start command prompt as admin and go to your project’s folder:

D:\000 WORK 2017/2 general/00 2018 TEVIN/OUR ETR/CharmSources

There run

**Git init this will create an empty git repository**

**git add . this will add everything (.) it finds within this folder into the repository**

This worked and AUTOMATICALLY pycharm recognized now my /CharmSources/ folder as git repo.

### No branch was found – only the master exists after this

I could not download from github anything and going to git > branches only the master was shown – the option to create a branch was not active????

Finally, I used the VCS > Import into version control > share on github

This was successful: this means I created ANOTHER master repository at github, NOT the one I created with DELL pycharm….

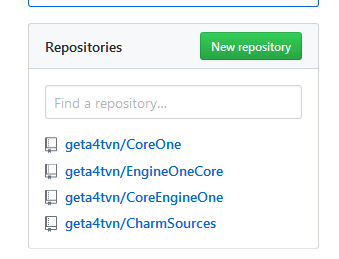
### Checkout online github

Login to github.com to my account and I see 4 FOUR repositories!

Latest was charmsources, the repository created just now for TurboX pycharm…

So, using the SAME account at github doesn’t mean you automatically share the SAME repository…

Of course, you can make as many repositories as you like…



So I need to find out how to use github and create branches on which to work … and then merge into …

## Adding the 3rd module, Execution

### How to pass a class object from one module (the CheckIn.py) to another (the execution.py)

I tested this by choosing to pass the SystemState class from CheckIn.py to Execution.py

#### NO NEED to Repeat same class definition in execution.py

I started execution.py by copying from CheckIn.py the same class definition **class SyState(object)**

But this IS NOT NEEDED as I commented the whole thing in execution.py and still I get correct operation.

This is because in CheckIn.py I declare SystemState as global and I repeat that inside the definition of the ektelese() function

#### NO NEED to repeat connections to static and journal.sqlite in execution.py

It’s enough that connection and cursors are created in CheckIn.py !!!

Perhaps because I DO NOT insert to db directly in Execution.py but indirectly, by calling the function I setup for class SystemState. Here is the call I made:

def ektelese**(**x**)**:  
 global SystemState  
 SystemState=x  
 print**('------- we are in execution.ektelese()')** print**('OpMode',**SystemState**.**OpMode**)** SystemState**.**SaveCurStat**()** print**('I just executed a function belonging to a class from another module!!!!!!')** print**('Time Stamp is',**SystemState**.**OpenRcptTStamp**)**

PERFECT!

The SystemState.SaveCurStat() is working from within the ektelese() function even though it is defined inside the SyState(object) class

class SyState**(**object**)**:  
 def \_\_init\_\_**(**self**,** OpMode**,** OpenRcpt**,** OpenRcptNum**,** OpenRcptTStamp**,** OpenDay**,** TStamp**,** LastZTStamp**,** LastZ**,** ShutDn**,** Poff**,** BatLow**,** RTCLow**,** FMerr**,** \  
 ClerkPerms**,** ClerkCode**,** PrnStat**,** WiFiStat**,** NetStat**,** DemoMode**,** Zpending**)**: # 20 columns  
 self**.**OpMode = OpMode # 0 = in me

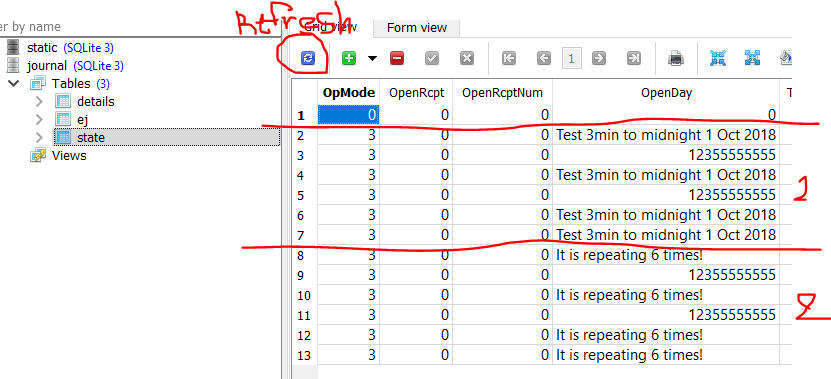
#### The whole object SystemState is passing through!

So actually what we need is ektelese(x,y,z) where x is the SyState object, y is the ReceiptId object and z is the Phrase object!

## First debug run shows 6 times writes to db instead of 1 expected!

That explains perhaps why THE WHOLE THING TOOK 500-700 msec to execute!!! (which scares me)

I found out about it by checking what is written into the state table in journal.sqlite:



#### End of work, go VCS > Commit and Push to update github

#### A PULL REQUEST is when you want to merge your branch with master, it is NOT about making a copy from github to local repo

# 5 October 2018

## CheckIn.py: how it works

### FIRST you read SystemState from journal.sqlite, table state

The engine status is described in class SyState

The object that is created is SystemState and contains even hardware errors data, I don’t know if this is the place to have but anyway….

**Basic flags to use:**

**SystemState.OpMode** is 0 when in normal receipt operation, 1 in reports, 2 in programming

**SystemState.OpenRcpt** is 0 when no open receipt and 1 when we are into a receipt

* When SystemState.OpenRcpt=0 the last receipt is ONE ROW in journal.sqlite, table journal
* When SystemState.OpenRcpt=1 all the lines of the receipt that is OPEN are in journal.sqlite, table details so Execution.py may use these lines to continue with calculating total, signatures,

This is the logic:

* Machine is turned ON
* When first keystroke arrives (OR execute an initiliaze script in InputKeys.py) the SystemState object is read from journal.sqlite / state **LINE 100-102 in CheckIn.py**
* As long as we have power and operate normally, this object is just in memory, we DO NOT write every change in db, we can’t write 1000s of times to flash every day
* When **power fails** we need to INTERRUPT Python using one of the I/O pins.
* **Power Fails** will be monitored by a monitor circuit that will trigger when external battery is disconnected OR when battery voltage gets too low. From that time on, there is a SECONDARY SUPER CAPACITOR that gives 15-20 sec of operation time for Linux.
* **Power Fail signal** will need to be asynchronously trigger our PYTHON – I don’t know how but we need to immediately start saving status to db and then start a power down through Linux system.
* **System Console** is available to be driven by one of the slave microcontrollers, but EVERYTHING can be driven by bash script and python INTERNALLY from Linux system, NO NEED to use system console to power down system, so we need system console perhaps to do remote upgrades??? – I don’t know how to use sys console
* **When Powering OFF,** battery power is there so we have plenty of time to save state and power down linux without power fail. However, POWER OFF procedure is same as PFAIL

## The journal.sqlite

Journal.sqlite is used dynamically, writing and reading all the time with operational data.

There is a second db in the system, static.sqlite, which contains data that DO NOT change all the time like PLU and HEADER tables. Static db will be only written using PROGRAMMING menu

### Static.sqlite with tables like plu or clerks etc DOES NOT CHANGE and this means that reports…

Reports will be DYNAMICALLY created by going through the journal tables.

**So, if a PLU is sold, the sale will NOT be recorded into plu table AT ALL.**

The ONLY record will be in the InvLine table, that is the line that contains the details of the sail, that is price and quantity at the moment of sale.

To get a PLU SALES report, the system will go through the InvLine table and will add up all the records there.

This means that we can freely change / edit / add any PLU or department and still be able to draw accurate reports because the changed PLU or DPT will not affect or delete the LINE (log) record of the past transaction. IDEA: add a “CHANGED” column in PLU to flag when the relative PLU is edited, changed or deactivated.

There are 3 tables in journal db:

**State:**

We only need one row in this table, holding the last state saved when device was powered off or when a power fail happens. Perhaps, to avoid deleting every time a new state is stored, we can just keep adding to this table and only take out the latest row. STATE will be written ONLY at power offs and power fails, so we will not abuse the table with a lot of rows…

**Details:**

Each row here is a row of the current OPEN receipt or invoice

When a receipt or invoice is CLOSED then we can delete all rows in details

Ej:

Each row here is a single receipt or invoice. This will be the biggest table of the device. Each row will contain ALL \_a text.

The idea here is to NOT PRODUCE

# The static.sqlite database

This contains things that don’t change much.

We have the following tables (modified 7 October 2018 to be used for retail receipts, invoices, credit notes and be compatible with KENYA TIMS)

## List of all tables in static.sqlite

### Items and Departments group

This group has