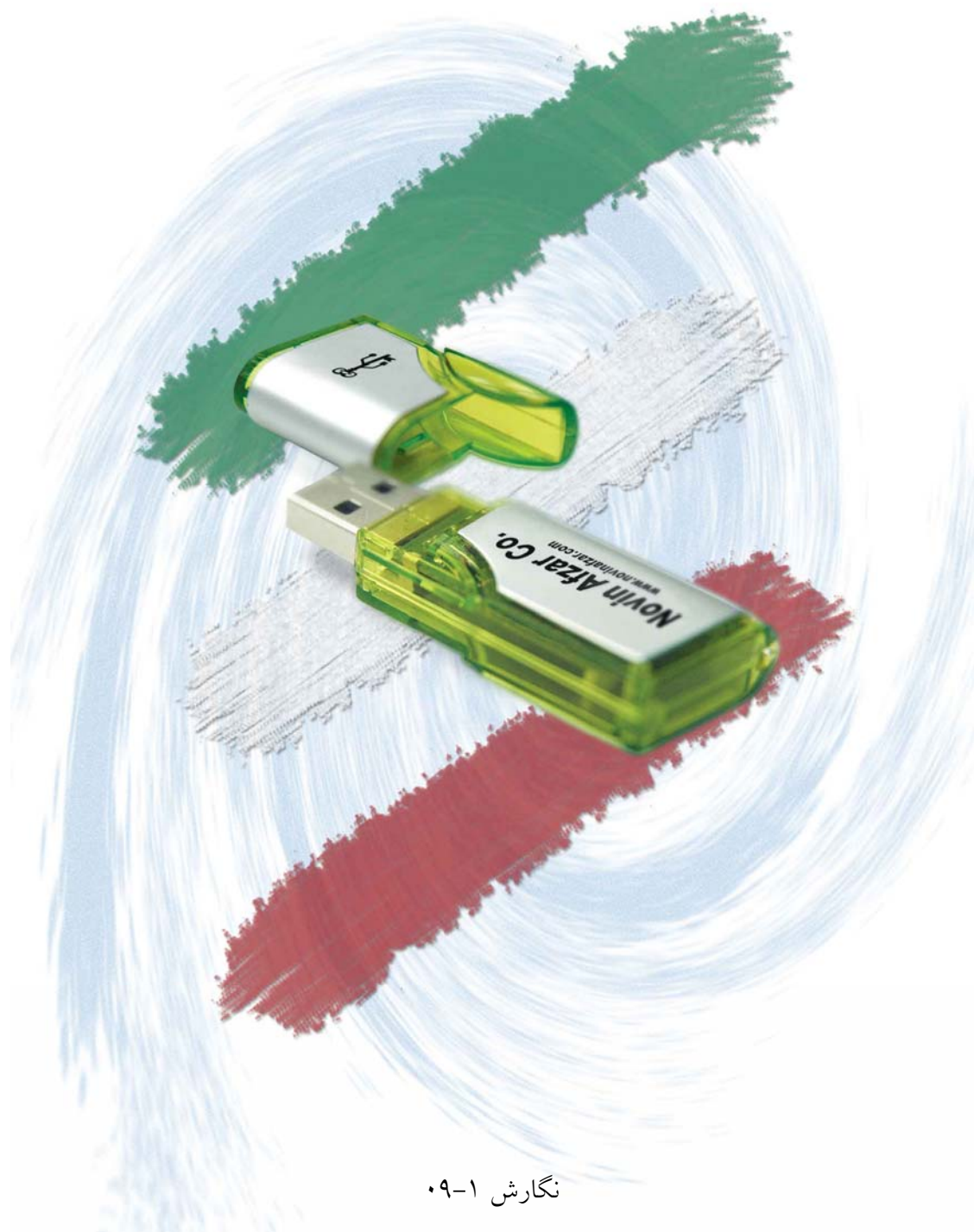


راهنمای استفاده از قفل سخت افزاری نوین

«HID نسخه ۸»



نگارش ۰۹-۱

زمستان ۱۳۸۷

فهرست

فصل اول: مشخصات قفل نوین.....	۱
۱-۱- تنظیمات سمت تولید کننده.....	۱
۱-۱-۱- شماره سریال قفل.....	۱
۱-۱-۲- کد اختصاصی نمایندگی یا مشتری (VID: Vendor ID).....	۱
۲-۱- تنظیمات مشتری.....	۲
۱-۲-۱- رمز عبور مدیریت (Admin Password).....	۲
۲-۲-۱- رمز عبور برنامه نویس (Developer Password).....	۲
۳-۲-۱- بخش داده آزاد.....	۳
۴-۲-۱- کد محصول (PID).....	۳
۵-۲-۱- رمزنگاری متقارن AES.....	۳
۶-۲-۱- شمارنده/زمانسنج.....	۳
۷-۲-۱- مشخصه شبکه.....	۴
فصل دوم: نرم افزار برنامه ریزی (Builder).....	۵
۱-۲- ساختار ظاهری.....	۵
۲-۲- روال یک: تولید فایل تنظیمات قفل (NLC).....	۷
۳-۲- روال دو: تغییر فایل تنظیمات قفل (NLC).....	۱۱
۴-۲- روال سه: برنامه ریزی قفل با (NLC).....	۱۴
۵-۲- روال چهار: برنامه ریزی دستی (مرحله به مرحله).....	۱۶
۶-۲- ریست/فعالسازی.....	۲۰
فصل سوم: کتابخانه رابط (dll).....	۲۱
۱-۳- ارتباط با قفل در برنامه نویسی.....	۲۱
۱-۱-۳- متدهای رابط محلی (قفل تک کاربره).....	۲۱
۲-۱-۳- متدهای رابط شبکه (قفل چندکاربره).....	۲۳
۳-۱-۳- توضیح متدهای قفل.....	۲۵
پیوست ۱: کد خطاها.....	۴۰
پیوست ۲: توصیه های امنیتی.....	۴۲
پیوست ۳: قفل سخت افزاری تحت شبکه.....	۴۳



فصل اول: مشخصات قفل نوین

قفل نوین در واقع یک میکروکنترلر کوچک هست که از طریق پورت USB به کامپیوتر وصل می‌شود و این قفل یک نرم افزار داخل میکروکنترلر خود دارد که سرویس های مختلفی را در اختیار کامپیوتر قرار می‌دهد.

برای تنظیم سرویسها پارامترهای (ثابت ها) مختلفی داخل قفل تنظیم می‌شود که از لحاظ روال تنظیم آنها می‌توان در دو دسته طبقه بندی کرد.

۱-۱- تنظیمات سمت تولید کننده

دسته ای از تنظیمات را شرکت تولید کننده (نوین افزار) روی قفل تنظیم می‌کند که این پارامترهای فقط توسط تولید کننده قابل ایجاد و دستکاری می‌باشد.

۱-۱-۱- شماره سریال قفل

این عدد توسط شرکت سازنده به قفل اختصاص می‌یابد و ساختاری شبیه آدرس IP شبکه را دارد. شماره سریال یک عدد یکتا است. یعنی هیچ دو قفلی وجود ندارد که شماره سریال یکسان داشته باشند. در شماره سریال (به طور مثال 9.0.0.1) اولین قسمت سال تولید آنرا نشان می‌دهد. بطور مثال ۹ نشاندهنده سال ۲۰۰۹ می‌باشد.

۱-۱-۲- کد اختصاصی نمایندگی یا مشتری (VID: Vendor ID)

برای هر استفاده کننده یعنی افرادی که قفل خام تهیه می‌کنند، یک کد اختصاصی یا VID توسط شرکت سازنده بطور رایگان تخصیص داده می‌شود و در تمام مراحل استفاده از قفل این کد لازم است، این کد از سه یا چهار عدد بین (۰-۲۵۵) که با نقطه از هم جدا شده اند (شبیه به آدرس IP شبکه) تشکیل شده است. این کد جزو سطح امنیت قفل ها به حساب آمده و فقط توسط شرکت سازنده قابل اختصاص است. پس از دریافت اولین سری از قفل های خریداری شده، در مراحل بعدی برای دریافت قفل های خام با همان VID قبلی مدارک قانونی شخص حقیقی یا حقوقی خریدار لازم است. بدین گونه قفل ها با VID مشخص فقط به صاحب قانونی آن VID تحویل داده می‌شود. به طور معمول یک شماره VID به هر مشتری اختصاص می‌یابد ولی مشتری می‌تواند بنا به درخواست خویش چندین VID را برای خود اختصاص دهد.

در صورتی که طرف خریدار مشتری اصلی قفل باشد VID چهار قسمتی ارائه می‌شود ولی VID مربوط به قفل هایی که به صورت نمایندگی تولید کننده هستند سه قسمتی می‌باشد که قسمت چهارم



توسط نماینده تنظیم و به مشتریان تحویل می گردد. توجه داشته باشید که سه قسمت اول VID (از سمت چپ) ریشه^۱ و قسمت آخر زیر VID^۲ نامیده می شود.

۱-۲- تنظیمات مشتری

این تنظیمات شامل پارامترهایی می شود که تغییر یا تنظیم آن فقط توسط سطح دسترسی مدیریت قفل امکانپذیر است و پس از تنظیم سمت کاربر نهایی غیرقابل تغییر است.

۱-۲-۱- رمز عبور مدیریت (Admin Password)

سطح دسترسی برای تنظیم کردن یا تغییر پارامترهای قفل رمز عبور Admin است که حداکثر ۱۶ رقم می باشد. به دلایل امنیتی، این سطح دسترسی را نباید به غیر از نرم افزار برنامه ریز قفل (Builder) در جای دیگر استفاده کرد. همچنین نباید در اختیار سایر افراد مانند برنامه نویسیها قرار داد. این رمز را حتما به خاطر بسپارید، زیرا که در صورت disable شدن قفل، این رمز برای ریست کردن یا فعالسازی قفل لازم است. همچنین در صورت نیاز به ریست کردن رمز عبور کاربر (PIN Code) داشتن رمز عبور مدیریت الزامی است.

۱-۲-۲- رمز عبور برنامه نویس (Developer Password)

برای استفاده از متدهای قفل در برنامه نویسی از این سطح امنیت می توان استفاده کرد. که ساختار آن مانند رمز عبور قبلی می باشد. در خواندن یا نوشتن قسمت داده یا رمزنگاری AES اصلی بودن این رمز عبور نیاز است.

نکته ۱: برای رمزهای عبور مدیریت و برنامه نویس حتما باید دو کلمه مجزا استفاده کرد.

نکته ۲: توجه داشته باشید چهار بار استفاده مکرر از رمز عبور و VID نادرست باعث می شود قفل disable شود. یعنی زمانی که VID، رمز عبور مدیریت یا برنامه نویس چهار بار یا رمز عبور کاربر نهایی چهار بار اشتباه زده شود این اتفاق می افتد. برای خروج از این حالت می توان قفل را با داشتن عبور مدیریت توسط نرم افزار برنامه ریزی ریست کرد. در صورتی که این عمل نیز چهار بار با رمز عبور مدیریت یا VID اشتباه انجام گردد قفل به حالت مسدود (Blocked) رفته و خروج از این حالت فقط توسط شرکت سازنده طی روال گارانتی امکانپذیر است.

نکته ۳: در صورتی که رمزهای عبور را در برنامه نویسی استفاده می کنید از پاس کردن این کلمات به صورت رشته ساده به سرویسهای قفل خودداری کنید. زیرا که فرم ارسال به سرویسها به صورت

^۱ Root VID

^۲ Sub VID



suspend می‌توان قفل یا فقط قسمت Timer/Counter را با استفاده از سطح دسترسی admin ریست کرد. روش دیگری که برای اینکار مخصوصاً از راه دور مناسب است، استفاده از Activation Code است. این رشته که یک رشته ۳۲ کاراکتری است به دو روش قابل تولید است. هنگامی که قفل در اختیار admin باشد با استفاده از سطح دسترسی admin قابل ایجاد است. علاوه بر این با داشتن شماره سریال قفل، با استفاده از قفل دیگری که کلید AES اول آنها با هم یکسان است نیز قابل تولید است.

۱-۲-۷- مشخصه شبکه

در صورتی که بخواهیم قفل در شبکه به اشتراک گذاشته شود، باید حداکثر تعداد کاربران مجاز استفاده کننده از قفل را با استفاده از این پارامتر تعیین کرد. این عدد می‌تواند یکی از اعداد ۱ تا ۲۵۵ بوده یا در حالت بینهایت (Unlimited) تنظیم شده باشد.



فصل دوم: نرم افزار برنامه ریزی (Builder)

این نرم افزار برای مدیریت یا سرپرست سیستمی طراحی شده است که می‌خواهد قفل‌ها را جهت استفاده در نرم افزار یا وب سایت خاصی آماده کند.

۱-۲- ساختار ظاهری



از نظر ظاهری این نرم افزار شامل بخشهای زیر است:

۱- بخش لیست دستگاه‌ها و مشخصات دستگاه انتخاب شده: در این بخش لیستی شامل شماره سریال و نگارش قفل‌های متصل به کامپیوتر نمایش داده می‌شود. با انتخاب هر یک از دستگاه‌های موجود در لیست مشخصات دستگاه انتخاب شده در زیر لیست نمایش داده می‌شود. علامت ✓ نشان دهنده ثبت شدن آن پارامتر در قفل و ✗ نشان دهنده خام (خالی) بودن آن پارامتر در قفل است.

علاوه بر این مشخصات دستگاه انتخاب شد (شماره سریال / نگارش / وضعیت) در قسمت نوار وضعیت (StatusBar) نمایش داده می‌شود.

۲- قسمت فرم‌های مراحل مختلف کار که در وسط صفحه می‌باشد: در مراحل مختلف کار تنظیم و انتخاب‌های متناسب با نوع کار نمایش داده می‌شود.

۳- کلیدهای کنترل مرحله که شامل سه کلید «صفحه اول»، «قبلی» و «بعدي» است، مراحل حرکت بین آنها را کنترل می‌کند.



۴- قسمت راهنما که با کلید بیشتر باز و بسته می‌شود.

۵- نوار وضعیت که وضعیت قفل انتخاب شده را نمایش می‌دهد.

توجه داشته باشید قسمت سوم این نوار، وضعیت کاری قفل انتخاب شده را نمایش می‌دهد. با انتخاب وسیله سه حالت احتمالی برای قفل را نشان می‌دهد.

حالت اول: «فعال» که نشان دهنده وضعیت نرمال کاری است .

حالت دوم: «غیر فعال» نشان دهنده غیر فعال بودن قفل را دارد که مشخص کننده استفاده نادرست از سطوح دسترسی قفل این وضعیت پیش آماده است .

حالت سوم: «مسدود بودن» قفل است که این نیز نشان دهنده استفاده نادرست از سطح

دسترسی مدیریت در هنگام فعال سازی یا ریست کردن قفل می‌باشد. توجه داشته باشید در صورتی حالت دوم تشخیص داده شود. نرم افزار به طور خودکار فرم فعالسازی قفل را باز می‌کند که با داشتن VID و رمز عبور مدیریت می‌توان قفل را فعال کرد یا با زدن کلید ریست به حالت خام اولیه درآورد.

برای سهولت دسترسی به قسمت‌های مختلف برنامه کلیدهای میانبری نیز در نرم افزار وجود دارد که به شرح زیر است:

F1: راهنما	F2: شروع کار
F3: ریست و فعال سازی	F10: خروج
Alt+1: روال یک	Alt+2: روال دو
Alt+3: روال سه	Alt+4: روال چهار

برای شروع کار با این نرم افزار پس از زدن شروع کار فرم زیر دیده می‌شود.



چنانچه مشاهده می‌کنید چهار روال اصلی برای کار با این نرم افزار وجود دارد. همچنین یک روال جانبی برای ریست کردن یا فعالسازی قفل نیز از طریق همین فرم در دسترس می‌باشد.



۲-۲- روال یک: تولید فایل تنظیمات قفل (NLC)

تولید فایل NLC که برای ذخیره پارامترهای مورد نظر در یک فایل استفاده می شود. فایل NLC محتوی پارامترهای قفل می باشد که اطلاعات در این فایل بصورت رمز شده نگهداری می شوند .

- مرحله ۱ :

در فرم بالا باید VID قفل و رمز اولیه مدیریت قفل را باید وارد کنید. توجه داشته باشید که VID رشته ای شبیه آدرس IP هست که از طرف شرکت سازنده به مشتری اختصاص داده می شود. این کد برای تمام قفل های یک مشتری برای همه مراحل خرید ثابت می باشد مگر اینکه برای تغییر آن درخواستی از طرف مشتری شود. این VID در اولین بار خرید تکی به صورت آزمایشی (عمومی) ارائه می شود و در مراحل بعدی خرید کد اختصاصی ارائه می شود . VID آزمایشی عبارتست از 109.232.151.192 که با کلیک کردن روی برچسب VID روی فرم فوق به صورت خودکار در محل VID تایپ می شود. رمز عبور قفل ها به صورت پیش فرض خالی می باشد .

- مرحله ۲ :



رمز عبور Admin قفل که حداکثر می تواند ۱۶ کاراکتر باشد برای برنامه ریزی قفل استفاده می شود و جایگاه استفاده آن فقط در نرم افزار برنامه ریزی قفل می باشد .
این رمز برای تغییر در پارامترهای قفل بعدا مورد استفاده قرار می گیرد و وجود آن به جلوگیری از دستکاری در پارامترهای قفل کمک می کند. همچنین دانستن آن برای Reset کردن یا خارج کردن قفل از حالت غیر فعال (disable) لازم است.

- مرحله ۳ :

رمز عبور برنامه نویسی همانطور که از اسمش پیداست جهت استفاده از برخی سرویسهای قفل در برنامه نویسی مورد استفاده قرار می گیرد.

این رمز برای سرویسهای زیر استفاده می شود:

- رمزنگاری AES اصلی (با استفاده از کلید اصلی) در `GetEncryption`
- خواندن قسمتی از حافظه قفل با استفاده از سرویسهای `GetByte`، `GetChar`، `GetBlockstr` و `GetBlockHexstr`
- نوشتن در قسمتی از حافظه قفل با استفاده از `SetByte`، `SetChar`، `SetBlockstr` و `SetBlockHexStr`
- رمزنگاری متقارن AES با استفاده از توابع `GetEncryption` و `GetDecryption`

- مرحله ۴ :

شماره نرم افزار یا کد محصول نیز همانند VID از چهار عدد تشکیل شده است که با نقطه از هم جدا شده اند. از این کد می توان برای شماره گذاری نسخه های نرم افزارها استفاده کرد.



- مرحله ۵ :

همانطور که قبلاً ذکر شد یکی از مهمترین قابلیت های موجود در قفل های نوین الگوریتم رمزنگاری AES است. این الگوریتم رمزنگاری براساس کلید ۱۲۸ بیتی عمل می کند، به عبارتی کلید از ۱۶ عدد (بین ۰-۲۵۵) که با نقطه از هم جدا شده اند تشکیل شده اند. از طریق فرم فوق می توانیم کلیدی برای این رمزنگاری تعیین کنیم.

- مرحله ۶ :

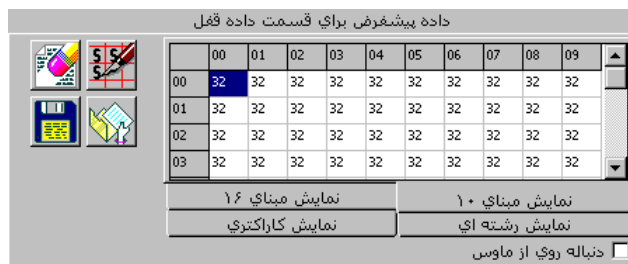
در قسمت مقدار شمارنده می توان عددی ۳ بیتی (حداکثر ۱۶,۷۷۷,۲۱۵) را به عنوان حداکثر مقدار شمارنده تعیین کرد، این عدد را می توان در مراحل مختلف استفاده توسط متد مربوطه کاهش داد یا با استفاده از فعال کردن زمانسنج (کاهنده خودکار) به طور خودکار از هر ۵,۵۹ ثانیه یک واحد کاهش داد.

- مرحله ۷ :







در صورتی که بخواهیم قفل در شبکه به اشتراک گذارده شود، علاوه بر فعال کردن این مشخصه می توان حداکثر تعداد کاربران مجاز استفاده کننده از قفل را با استفاده از این پارامتر تعیین کرد. این عدد می تواند یکی از اعداد ۱ تا ۲۵۴ بوده یا در حالت نامحدود (Unlimited) تنظیم کرد.

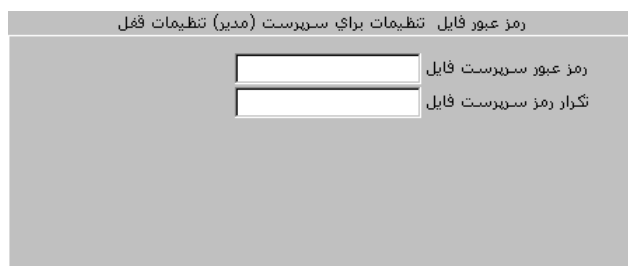
- مرحله ۸:



در این فرم می توانید داده پیش فرض قفل را تعریف کنید. این داده را که می تواند حداکثر ۴۰۰ کاراکتر باشد. می توان در این فرم به صورت رشته ای، کاراکتری یا به صورت کد اسکی در مینای ۱۰ یا ۱۶ دید. امکانات اضافی شامل:

- کلید  برای پر کردن خانه ها با یک عدد (کاراکتر) خاص استفاده می شود. توسط کلید  تمام خانه ها با کد ۳۲ (فاصله) پر می شود.
- با کلید  می توانید این داده را از فایل بخوانید یا با کلید  در فایل ذخیره کنید.

- مرحله ۹:



رمز عبور سرپرست فایل به هیچ یک از سطوح دسترسی قفل ربطی ندارد و فقط جهت دسترسی به پارامترهای ذخیره شده در فایل NLC مورد استفاده قرار می گیرد. توجه داشته باشید برای تغییر فایل NLC (روال دوم) داشتن این رمز عبور لازم است.



- مرحله ۱۰:

رمز عبور فایل تنظیمات برای اپراتور برنامه ریزی قفل

رمز عبور اپراتور برنامه ریزی

تکرار رمز عبور اپراتور

رمز عبور اپراتور برنامه ریزی نیز مانند رمز عبور سرپرست هیچ ربطی به رمز های قفل ندارد. این رمز عبور در هنگام استفاده از روال سوم (برنامه ریزی قفل با NLC) لازم است. توجه داشته باشید هنگامی که تعداد زیادی قفل را قرار است برنامه ریزی کنید می‌توانید چند اپراتور برای این کار اختصاص دهید و با در اختیار گذاشتن رمز عبور اپراتوری آنها می‌توانند برنامه ریزی اولیه قفل‌ها را بدون آنکه از پارامترهای تنظیم شده با خبر شوند انجام دهند.

- مرحله ۱۱:

ذخیره فایل تنظیمات برای استفاده های بعدی

فایل NLC:

تعیین فایل

در این باید فایل مقصد را برای ذخیره کردن تنظیمات انجام شده انتخاب نمایید. پسوند فایل تنظیمات NLC^۱ است که مخفف می‌باشد. در این فایل اطلاعات به صورت رمز شده ذخیره می‌شود که مشاهده یا استفاده از آن بدون داشتن رمز عبور فایل امکانپذیر نخواهد بود.

۲-۳- روال دو: تغییر فایل تنظیمات قفل (NLC)

تغییر فایل NLC جهت تغییر در پارامترهای ذخیره شده در یک فایل NLC مورد استفاده قرار می‌گیرد، برای تغییر فایل NLC داشتن رمز عبور سرپرست (Supervisor) فایل نیاز است.

^۱ Novin Lock Configuration



- مرحله ۱:

از طریق فرم فوق می‌توانید فایل مورد نظر خود را انتخاب کنید تا اطلاعات و تنظیمات از پیش ذخیره شده از آن فایل خوانده شود.
نوع فایل NLC می‌باشد که این فایل قبلاً توسط فقط همین نرم افزار باید تولید شده باشد.

- مرحله ۲:

برای دسترسی به اطلاعات ذخیره شده در فایل NLC رمز عبور فایل را باید وارد کنید.
در صورتی که می‌خواهید تنظیمات موجود در فایل را تغییر دهید (روال دوم) حتماً باید رمز عبور سرپرست فایل را داشته باشید و وارد کنید. ولی زمانی که می‌خواهید قفل‌ها را از روی فایل برنامه ریزی کنید (روال سوم) می‌توانید هم رمز عبور سرپرست را استفاده کنید و هم رمز عبور اپراتور برنامه ریزی را.
اگر رمزهای عبور فایل را ندارید آنرا باید از ایجاد کننده فایل دریافت کنید.

- مرحله های ۳ الی ۱۰:

تکرار مراحل ۱ تا ۱۰ روال یک (تولید فایل تنظیمات قفل (NLC))



- مرحله ۱۱:

نحوه ذخیره فایل تنظیمات تغییر یافته

ذخیره روی فایل اصلی (فایل باز شده تنظیمات)
 ذخیره تنظیمات روی فایل جدید
 ذخیره فایل با رمزهای عبور قدیمی فایل (بدون گرفتن رمز جدید)

پس از تغییرات مورد نظر در فایل باز شده می‌توانید این تغییرات را روی فایل قبلی یا در یک فایل جدید ذخیره کنید. بدین منظور فایل مورد نظر خود را برای ذخیره فایل انتخاب نمایید. در صورتی که می‌خواهید از رمزهای عبور فایل باز شده استفاده کنید می‌توانید گزینه ذخیره با رمزهای قدیمی را تیک نزیند که در غیر این صورت رمزهای عبور جدید برای ذخیره فایل در مراحل بعدی پرسیده خواهد شد.

- مرحله ۱۲:

رمز عبور فایل تنظیمات برای سرپرست (مدیر) تنظیمات قفل

رمز عبور سرپرست فایل
 تکرار رمز سرپرست فایل

رمز عبور سرپرست فایل به هیچ یک از سطوح دسترسی قفل ربطی ندارد و فقط جهت دسترسی به پارامترهای ذخیره شده در فایل NLC مورد استفاده قرار می‌گیرد. توجه داشته باشید برای تغییر فایل NLC (روال دوم) داشتن این رمز عبور لازم است.

- مرحله ۱۳:

رمز عبور فایل تنظیمات برای اپراتور برنامه ریزی قفل

رمز عبور اپراتور برنامه ریزی
 تکرار رمز عبور اپراتور



رمز عبور اپراتور برنامه ریزی نیز مانند رمز عبور سرپرست هیچ ربطی به رمز های قفل ندارد. این رمز عبور در هنگام استفاده از روال سوم (برنامه ریزی قفل با NLC) لازم است. توجه داشته باشید هنگامی که تعداد زیادی قفل را قرار است برنامه ریزی کنید می توانید چند اپراتور برای این کار اختصاص دهید و با در اختیار گذاشتن رمز عبور اپراتوری آنها می توانید برنامه ریزی اولیه قفل ها را بدون آنکه از پارامترهای تنظیم شده با خبر شوند انجام دهند.

- مرحله ۱۴:

ذخیره فایل تنظیمات برای استفاده های بعدی

فایل NLC:

تعیین فایل

در این باید فایل مقصد را برای ذخیره کردن تنظیمات انجام شده انتخاب نمایید. نوع فایل تنظیمات NLC است که مخفف Novin Lock Configuration می باشد. در این فایل اطلاعات به صورت رمز شده ذخیره می شود که مشاهده یا استفاده از آن بدون داشتن رمز عبور فایل امکانپذیر نخواهد بود.

۲-۴- روال سه: برنامه ریزی قفل با (NLC)

برنامه ریزی قفل با استفاده از NLC برای انتقال پارامترهای تنظیم شده در یک فایل NLC به قفل، مورد استفاده قرار می گیرد.

- مرحله ۱:

بازکردن فایل تنظیمات قفل به فرمت NLC

فایل NLC:

بازکردن فایل



از طریق فرم فوق می‌توانید فایل مورد نظر خود را انتخاب کنید تا اطلاعات و تنظیمات از پیش ذخیره شده از آن فایل خوانده شود.

نوع فایل NLC می‌باشد که این فایل قبلاً توسط فقط همین نرم افزار باید تولید شده باشد.

- مرحله ۲:

برای دسترسی به اطلاعات ذخیره شده در فایل NLC رمز عبور فایل را باید وارد کنید. در صورتی که می‌خواهید تنظیمات موجود در فایل را تغییر دهید (روال دوم) حتماً باید رمز عبور سرپرست فایل را داشته باشید و وارد کنید. ولی زمانی که می‌خواهید قفل‌ها را از روی فایل برنامه ریزی کنید (روال سوم) می‌توانید هم رمز عبور سرپرست را استفاده کنید و هم رمز عبور اپراتور برنامه ریزی را. اگر رمزهای عبور فایل را ندارید آنرا باید از ایجاد کننده فایل دریافت کنید.

- مرحله ۳:

برای انتقال تنظیمات خوانده شده از فایل NLC به داخل قفل باید از فرم فوق استفاده کرد. بدین منظور دو روال را می‌توان استفاده کرد.

- اجرای دستی: زمانی که تیک مورد اجرای خودکار زده نشده است، پس از انتخاب قفل مورد نظر از لیست قفل‌ها یافته شده کلید «نوشتن در دستگاه» را بدین منظور می‌توانید بزنید.
- اجرای خودکار: با تیک زدن اجرای خودکار با پیدا کردن اولین قفل نرم افزار شروع به برنامه ریزی آن می‌نماید. در صورتی که بیش از یک قفل وصل شود به ترتیب شناسایی، شماره سریال قفل‌ها در لیستی قرار می‌گیرد که به ترتیب آنها را برنامه ریزی کند.



- استفاده از این مورد برای افراد مبتدی پیشنهاد نمی شود مخصوصا زمانی که قفل ها فاقد LED نشاندهنده وضعیت فعالیت می باشد. زیرا که تشخیص اینکه کدام قفل در حال برنامه ریزی هست یا کدام قفل ها برنامه ریزی شده اند اندکی دشوار است.
 - می توانید قفل ها را به صورت گروهی به سیستم وصل کنید و تا اتمام برنامه ریزی همه قفل ها هیچ قفلی را از کامپیوتر نکشید و پس از اتمام کار می توانید با انتخاب تک تک قفل ها از برنامه ریزی شدن آنها مطمئن شوید و پس از کشیدن همه قفل از کامپیوتر، گروه بعدی را برنامه ریزی کنید.
- طی عملیات برنامه ریزی پیشرفت مراحل و نوع کار در حال انجام از طریق همین فرم کاملا مشهود است.

۲-۵- روال چهار: برنامه ریزی دستی (مرحله به مرحله)

برنامه ریزی مرحله به مرحله برای ثبت پارامترهای قفل انتخاب شده در هر مرحله مورد استفاده قرار می گیرد یعنی صفحه به صفحه پارامترها وارد و در قفل ثبت می گردد.

- مرحله ۱:

کد ویژه مشتری و رمز فعلی مدیریت	
<input type="text" value="0 . 0 . 0"/>	کد ویژه مشتری (VID):
<input type="text"/>	رمز عبور فعلی مدیریت:

در فرم بالا باید VID قفل و رمز اولیه مدیریت قفل را باید وارد کنید. توجه داشته باشید که VID رشته ای شبیه آدرس IP هست که از طرف شرکت سازنده به مشتری اختصاص داده می شود. این کد برای تمام قفل های یک مشتری برای همه مراحل خرید ثابت می باشد مگر اینکه برای تغییر آن درخواستی از طرف مشتری شود. این VID در اولین بار خرید تکی به صورت آزمایشی (عمومی) ارائه می شود و در مراحل بعدی خرید کد اختصاصی ارائه می شود. VID آزمایشی عبارتست از 109.232.151.192 که با کلیک کردن روی برچسب VID روی فرم فوق به صورت خودکار در محل VID تایپ می شود. رمز عبور قفل های خام به صورت پیش فرض خالی می باشد.



- مرحله ۲:

رمز عبور مدیریت قفل

رمز عبور مدیریت

تکرار رمز مدیریت

ثابت از نو

رمز عبور Admin قفل که حداکثر می تواند ۱۶ کاراکتر باشد برای برنامه ریزی قفل استفاده می شود و جایگاه استفاده آن فقط در نرم افزار برنامه ریزی قفل می باشد. این رمز برای تغییر در پارامترهای قفل بعدا مورد استفاده قرار می گیرد و وجود آن به جلوگیری از دستکاری در پارامترهای قفل کمک می کند. همچنین دانستن آن برای Reset کردن یا خارج کردن قفل از حالت غیر فعال (disable) لازم است.

- مرحله ۳:

رمز عبور برنامه نویسی با قفل

رمز عبور برنامه نویسی

تکرار رمز برنامه نویسی

ثابت از نو

رمز عبور برنامه نویسی همانطور که از اسمش پیداست جهت استفاده از برخی سرویسهای قفل در برنامه نویسی مورد استفاده قرار می گیرد.

این رمز برای سرویسهای زیر استفاده می شود:

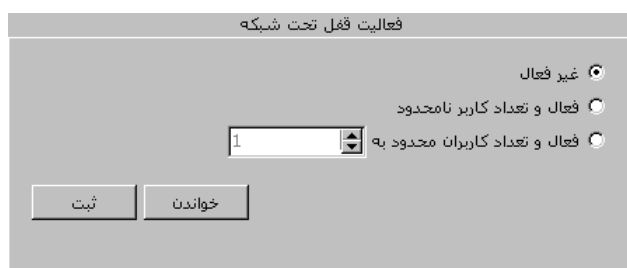
- خواندن قسمتی از حافظه قفل با استفاده از سرویسهای `GetBlockstr`، `GetChar`، `GetByte` و `GetBlockHexstr`
- نوشتن در قسمتی از حافظه قفل با استفاده از `SetBlockstr`، `SetChar`، `SetByte` و `SetBlockHexStr`
- رمزنگاری متقارن AES با استفاده از توابع `GetEncryption` و `GetDecryption`



در صورتی که پس از اتمام این محدودیت بخواهیم قفل را مجدداً فعال کنیم می‌توانیم از طریق تولید کلید فعالسازی و ارسال آن به قفل آنرا فعال کنیم. بدین منظور از طریق کلید تولید کد فعالسازی که علاوه بر این فرم در قسمت ریست/فعالسازی نیز وجود دارد می‌توان اقدام به تولید کد فعالسازی نمود.

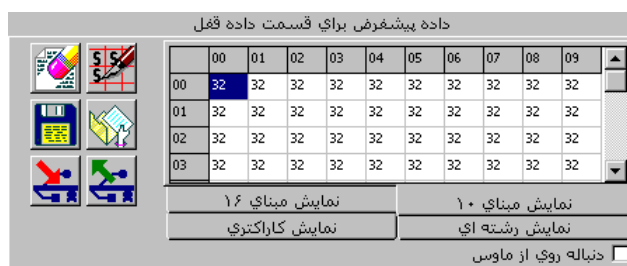
کلید خواندن وضعیت فعلی شمارنده/زمانسنج را از قفل استخراج کرده و کلید ثبت نیز تنظیمات مورد نظر را در قفل اعمال می‌کند.

- مرحله ۷:







در صورتی که بخواهیم قفل در شبکه به اشتراک گذاشته شود، علاوه بر فعال کردن این مشخصه می‌توان حداکثر تعداد کاربران مجاز استفاده کننده از قفل را با استفاده از این پارامتر تعیین کرد. این عدد می‌تواند یکی از اعداد ۱ تا ۲۵۴ بوده یا در حالت نامحدود (Unlimited) تنظیم کرد. کلید های خواندن و ثبت عملیات استخراج یا ذخیره تنظیمات مشخصه شبکه را انجام می‌دهند.



- مرحله ۸:



در این فرم می‌توانید داده پیش فرض قفل را تعریف کنید. این داده را که می‌تواند حداکثر ۳۲ کاراکتر باشد می‌توان در این فرم به صورت رشته ای، کاراکتری یا به صورت کد اسکی در مبنای ۱۰ یا ۱۶ دید. امکانات اضافی شامل:

- کلید  برای پر کردن خانه ها با یک عدد (کاراکتر) خاص استفاده می‌شود. توسط کلید  تمام خانه ها با کد ۳۲ (فاصله) پر می‌شود.
- با کلید  می‌توانید این داده را از فایل بخوانید یا با کلید  در فایل ذخیره کنید.



- در صورتیکه روال مرحله به مرحله را استفاده می‌کنید دو کلید نیز به فرم اضافه می‌شود. از طریق کلید  داده موجود در قفل انتخاب شده خوانده شده و با زدن  اطلاعات موجود در جدول داده پیش فرض به داخل قفل انتقال می‌یابد.

۲-۶- ریست/فعالسازی

فرم ریست/فعالسازی

در صورتی که قفل شما غیر فعال شده است و یا می‌خواهید آنرا به صورت خام در آورید و یا بین کد کاربر آنرا ریست (پاک) کنید، می‌توانید پس از وارد کردن سطح دسترسی مدیریت از طریق کلیدهای مربوطه خواسته خود را برآورده کنید.

کد ویژه مشتری (VID): 0 . 0 . 0

رمز عبور فعلی مدیریت:

فعال کردن قفل ریست کردن قفل کد فعالسازی انصراف

در صورتی که قفل شما غیر فعال شده است و یا می‌خواهید آن را به صورت خام در بیاورید، پس از وارد کردن VID و رمز عبور مدیریت از طریق کلیدهای مربوطه خواسته خود را بر آورده کنید. علاوه بر این با زدن کلید کد فعالسازی می‌توانید به بخش تولید این کد بروید.

ایجاد کد فعالسازی

کد ویژه مشتری (VID): 0 . 0 . 0

رمز عبور فعلی مدیریت:

ایجاد با استفاده از شناسه فعلی (انتخاب شده)

ایجاد با شناسه فعلی برای شماره سریال دیگر

شماره سریال قفل: 0 . 0 . 0

کد فعالسازی:

ایجاد کد فعالسازی انصراف

در این بخش شما می‌توانی برای قفل متصل به کامپیوتر که آنرا انتخاب کرده اید یا برای قفل دیگری که سریال آنرا می‌دانید کد فعالسازی تولید کرده و از طریق متد فعالسازی آنرا از وضعیت تعلیق در آورید. نکته: توجه داشته باشید زمانی می‌توانید برای قفل دیگر کد ایجاد کنید که کلید رمزنگاری AES قفل فعلی (انتخاب شده) را با کلید رمزنگاری شماره سریال دیگر یکسان تنظیم کرده باشید.



فصل سوم: کتابخانه رابط (dll)

۳-۱- ارتباط با قفل در برنامه نویسی

برای ارتباط با قفل نوین از طریق نرم افزارهایی که می نویسیم یا از طریق صفحه وب وجود فایل DLL رابط قفل لازم است. این فایل به دو صورت تحت نامهای XNovinHIDLock.dll و LNovinHIDLock.dll در CD مربوط به قفل موجود است. دسترسی به متدهای این دو فایل از طرق زیر امکانپذیر است.

۱. ارتباط با LNovinHIDLock.dll به صورت کتابخانه از طریق توابع export شده که می توان به طور مستقیم به DLL لینک شد و استفاده کرد.

۲. استفاده به صورت ActiveX از طریق فایل XNovinHIDLock.dll که قابل ثبت^۱ شدن در ویندوز است.

دو کلاس اتومات مشتق شده از NovinAfzar به نامهای clsHIDLock و clsNETLock در ActiveX وجود دارد ولی از طریق کتابخانه فقط می توان clsHIDLock را بازسازی کرد. برای فراخوانی قفل از داخل نرم افزار باید فایل مربوط به روتین های قفل (LNovinHIDLock.dll) را به پروژه اضافه شود.

راه دیگر ارتباط با قفل استفاده به صورت ActiveX، (clsHIDLock) است.

تعریف روالها

۳-۱-۱- متدهای رابط محلی (قفل تک کاربره)

روال های موجود در TclsHIDLock عبارتند از:

Methods:

Init	آماده سازی کلاس برای کار
GetFirstDevice	گرفتن مشخصات اولین قفل شناسایی شده
GetNextDevice	گرفتن مشخصات قفل بعدی شناسایی شده
GetDeviceCount	دریافت تعداد دستگاه های شناسایی شده
SelectDevice	انتخاب یک دستگاه برای اجرای دستورات

¹ Register



GetDeviceReady	تست وجود قفل
GetDeviceStatus	تست وضعیت قفل
GetSerial	دریافت شماره سریال قفل
GetVersion	دریافت شماره نسخه قفل
GetMemorySize	دریافت مقدار حافظه قابل دسترس
GetMemorySizeEx	دریافت مقدار کاراکتر قابل دسترس
SetDataByte	نوشتن یک بایت در حافظه قفل
SetDataBlockStr	نوشتن اطلاعات در حافظه بصورت رشته ای
SetDataHexBlock	نوشتن بلوک HEX در بخش داده
GetDataByte	خواندن یک بایت از حافظه قفل
GetDataBlockStr	خواندن اطلاعات از حافظه بصورت رشته ای
GetDataHexBlock	خواندن بلوک HEX از بخش داده
GetEncryption	رمزگذاری متقارن AES
GetDecryption	رمزگشایی متقارن AES
GetPID	گرفتن کد محصول
GetNET	خواندن مشخصه شبکه
ConvDelimitedStringToString	تبدیل رشته از DNString به رشته کاراکتری
ConvStringToDelimitedString	تبدیل رشته از رشته کاراکتری ^۱ به DNString ^۲
ConvHexStringToDelimitedString	تبدیل رشته HEX ^۳ به DNString
ConvDelimitedStringToHexString	تبدیل DNString به رشته HEX
ConvStringToHEXString	تبدیل رشته معمولی به رشته HEX
ConvHEXStringToString	تبدیل رشته HEX به رشته معمولی
GetTimerCounter	خواندن وضعیت شمارنده/زمانسنج
DecreaseCounter	کاهش شمارنده
Activate	فعالسازی قفل از طریق کد ویژه

Properties:

^۱ رشته کاراکتری یکی از سه نوع داده رشته ای است که به صورت معمولی (کاراکتری) و از نوع WideString است
^۲ رشته Delimited یا DNString از انواع دیگر داده رشته ای است که به صورت اعداد جدا شده با نقطه (همانند آدرس IP) است
^۳ نوع رشته ای HEX نوع داده ای است که کد اسکی کاراکترها به صورت اعداد پشت سر هم در مبنای ۱۶ می باشد



ErrNo	کد خطای حاصل از آخرین متد اجرا شده
ErrDescr	پیام خطای متناسب با کد خطا فوق (انگلیسی)
ErrDescrFA	پیام خطای متناسب با کد خطا فوق (فارسی)
UserPwd	رمز عبور سطح سه (کاربر)
DeviceName	نام وسیله در نتیجه جستجو
DeviceVer	نگارش وسیله در نتیجه جستجو
DeviceSerial	شماره سریال وسیله در نتیجه جستجو
SelectedName	نام وسیله انتخاب شده
SelectedVer	نگارش وسیله انتخاب شده
SelectedSerial	شماره سریال وسیله انتخاب شده

۳-۱-۲- متدهای رابط شبکه (قفل چندکاربره)

روال های موجود در TclsNETLock عبارتند از:

Methods:

Start	شروع اتصال به سرور
Stop	پایان اتصال به سرور
GetConnectionStatus	گرفتن وضعیت اتصال به سرور
GetDeviceReady	تست وجود قفل
GetDeviceStatus	تست وضعیت قفل
GetSerial	دریافت شماره سریال قفل
GetVersion	دریافت شماره نسخه قفل
GetMemorySize	دریافت مقدار حافظه قابل دسترس
GetMemorySizeEx	دریافت مقدار کاراکتر قابل دسترس
SetDataByte	نوشتن یک بایت در حافظه قفل
SetDataBlockStr	نوشتن اطلاعات در حافظه بصورت رشته ای
SetDataHexBlock	نوشتن بلوک HEX در بخش داده
GetDataByte	خواندن یک بایت از حافظه قفل
GetDataBlockStr	خواندن اطلاعات از حافظه بصورت رشته ای
GetDataHexBlock	خواندن بلوک HEX از بخش داده



GetEncryption	رمز گذاری متقارن AES
GetDecryption	رمز گشایی متقارن AES
GetPID	گرفتن کد محصول
GetNET	خواندن مشخصه شبکه
ConvDelimitedStringToString	تبدیل رشته از DNString به رشته کاراکتری
ConvStringToDelimitedString	تبدیل رشته از رشته کاراکتری به DNString
ConvHexStringToDelimitedString	تبدیل رشته HEX به DNString
ConvDelimitedStringToHexString	تبدیل DNString به رشته HEX
ConvStringToHEXString	تبدیل رشته معمولی به رشته HEX
ConvHEXStringToString	تبدیل رشته HEX به رشته معمولی
GetTimerCounter	خواندن وضعیت شمارنده/زمانسنج
DecreaseCounter	کاهش شمارنده
Activate	فعالسازی قفل از طریق کد ویژه
Properties:	
ErrNo	کد خطای حاصل از آخرین متد اجرا شده
ErrDescr	پیام خطای متناسب با کد خطا فوق (انگلیسی)
ErrDescrFA	پیام خطای متناسب با کد خطا فوق (فارسی)
Host	آدرس اسمی یا IP سرور قفل
Port	پورت سرور قفل



۳-۱-۳- توضیح متدهای قفل

- موارد اختصاصی TclsHIDLock:

Init	آماده سازی کلاس برای کار
procedure Init;	
برخی مواقع هنگام استفاده از DLL رابط شناسایی HID درست آماده سازی نمی شود و به خطای nErr_HID برخورد می کنید. برای رفع این مشکل باید هنگام اتصال به کتابخانه این تابع را فراخوانی کنید	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد	

GetFirstDevice	گرفتن مشخصات اولین قفل شناسایی شده
procedure GetFirstDevice;	
با توجه به اینکه کتابخانه قابلیت کار با بیش از یک دستگاه وصل شده به کامپیوتر را داراست. این روال مشخصات اولین دستگاه شناسایی شده را از طریق DeviceSerial، DeviceVer و DeviceName نمایش می دهد.	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد	

GetNextDevice	گرفتن مشخصات قفل بعدی شناسایی شده
procedure GetNextDevice;	
با این روتین مشخصات دستگاه بعدی شناسایی شده نمایش داده می شود.	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد	



GetDeviceCount	دریافت تعداد دستگاه های شناسایی شده
Function GetDeviceCount: Word;	
این تابع تعداد دستگاههای شناسایی شده سازگار با کتابخانه را نمایش می دهد.	
هیچ پارامتری ندارد	
مقدار بازگشتی: تعداد دستگاهها	

SelectDevice	انتخاب یک دستگاه برای اجرای دستورات
Procedure SelectDevice(SerialNo: WideString);	
برای استفاده از توابع که در ادامه توضیح داده می شود باید دستگاهی انتخاب شود که از طریق این روال انجام می پذیرد.	
در صورت عدم استفاده از این تابع حین کار به خطای nErr_DNS برخورد خواهید کرد.	
SerialNo : شماره سریال دستگاهی که می خواهید انتخاب کنید.	
هیچ مقدار بازگشتی ندارد	

DeviceName	نام وسیله در نتیجه جستجو
property DeviceName: WideString	
پس از استفاده از GetFirstDevice یا GetNextDevice این property نام وسیله شناسایی شده را نمایش می دهد.	
این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع widestring بر میگرداند	

DeviceVer	نگارش وسیله در نتیجه جستجو
property DeviceVer: WideString	
پس از استفاده از GetFirstDevice یا GetNextDevice این property نگارش وسیله شناسایی شده را نمایش می دهد.	
این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع widestring بر میگرداند	



DeviceSerial	شماره سریال وسیله در نتیجه جستجو
property DeviceSerial: WideString	
پس از استفاده از GetFirstDevice یا GetNextDevice این property شماره سریال وسیله شناسایی شده را نمایش می دهد.	
این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع widestring بر می گرداند	

SelectedName	نام وسیله انتخاب شده
property SelectedName: WideString	
پس از استفاده از SelectDevice این property نام وسیله شناسایی شده را نمایش می دهد.	
این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع widestring بر می گرداند	

SelectedVer	نگارش وسیله انتخاب شده
property SelectedVer: WideString	
پس از استفاده از SelectDevice این property نگارش وسیله شناسایی شده را نمایش می دهد.	
این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع widestring بر می گرداند	

SelectedSerial	شماره سریال وسیله انتخاب شده
property SelectedSerial: WideString	
پس از استفاده از SelectDevice این property شماره سریال وسیله شناسایی شده را نمایش می دهد.	
این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع widestring بر می گرداند	



- موارد اختصاصی TclsNETLock:

Start	شروع ارتباط با سرور
procedure Start;	
پس از تعیین آدرس و پورت سرور این متد ارتباط به سرور مورد نظر را برقرار می‌کند.	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد	

Stop	قطع ارتباط با سرور
procedure Stop;	
ارتباط مورد نظر با سرور را قطع می‌نماید	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد	

Host	آدرس IP سرور
property Host: WideString	
تعیین می‌کند سرور در کدام آدرس شبکه قرار دارد مثلا 192.168.0.1	
این مشخصه به صورت خواندنی و نوشتنی عمل می‌کند	

Port	پورت سرویس دهی Netserver
property Port: Word	
تعیین می‌کند کدام پورت سرور برای سرویس‌دهی تعیین شده است.	
این مشخصه به صورت خواندنی و نوشتنی عمل می‌کند	



- موارد مشترک TclsNETLock و TclsHIDLock:

GetDeviceReady	تست وجود قفل
<code>procedure GetDeviceReady;</code>	
ساده ترین روش برای چک کردن قفل در پورت USB.	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد	

GetDeviceStatus	تست وضعیت قفل
<code>procedure GetDeviceStatus;</code>	
تست وضعیت: فعال یا یکی از وضعیتهای غیرفعال (Disabled, Blocked or Catastrophic Disabled)	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد. وضعیت موجود قفل در ErrNo با خطاهای متناسب برگشت می‌شود.	

GetSerial	دریافت شماره سریال قفل
<code>function GetSerial: WideString;</code>	
هر قفل دارای یک سریال یکتا می‌باشد که در موقع تولید، توسط تولید کننده در قفل ذخیره می‌گردد با استفاده از این روتین می‌توان این سریال را بدست آورد.	
هیچ پارامتری ندارد	
مقدار برگشتی : در صورت اجرای موفقیت آمیز شماره سریال را که از نوع WideString می‌باشد باز می‌گرداند.	



GetVersion	دریافت شماره نسخه قفل
function GetVersion: WideString;	
هر قفل دارای نرم افزارهای داخلی می باشد که با توسعه قفل این نرم افزارها تغییر می کند. لذا هر قفل دارای شماره نسخه مربوط به خود است که از طریق این روتین می توان آنرا بدست آورد.	
هیچ پارامتری ندارد	
مقدار برگشتی : در صورت اجرای موفقیت آمیز شماره نسخه از نوع WideString برگشت داده می شود.	

GetMemorySize	دریافت مقدار حافظه قابل دسترس
function GetMemorySize: Word;	
با توجه به نوع قفل، حافظه قابل دسترس متغیر می باشد، با استفاده از این سرویس، می توان مقدار این حافظه را به بایت بدست آورد.	
هیچ پارامتری ندارد	
مقدار برگشتی : از نوع WORD می باشد و بعد از اجرا مقدار حافظه قابل دسترس را برگشت می دهد.	

GetMemorySizeEx	دریافت مقدار کاراکتر قابل دسترس
function GetMemorySizeEx: WideString;	
با توجه به نوع قفل، حافظه قابل دسترس و charset فضای داده ، متغیر می باشد، با استفاده از این سرویس، می توان مقدار این حافظه را برحسب کاراکتر و نوع چارست بدست آورد.	
هیچ پارامتری ندارد	
مقدار برگشتی : از نوع Widestring می باشد، تعداد کاراکترهای حافظه و نوع charset را برگشت می دهد.	



SetDataByte	نوشتن یک بایت در حافظه قفل
<pre>procedure SetDataByte(const sVID: WideString; const sPWD: WideString; Address: Word;DataByte: Byte);</pre>	
<p>برای نوشتن اطلاعات در قفل بصورت بایت به بایت از این روتین استفاده می‌شود این روتین در آدرس مشخص شده اطلاعات یک بایت را می‌نویسد.</p>	
<p>sVID: این پارامتر VID می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و باید بصورت WideString باشد.</p> <p>sPWD: این پارامتر رمز عبور قفل می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد.</p> <p>Address: آدرس مورد نظر جهت نوشتن اطلاعات و از نوع Word می‌باشد.</p> <p>DataByte: داده مورد نظر در این پارامتر ذخیره و بعد از اجرای روتین ، در آدرس مشخص شده ذخیره می‌گردد. نوع آن byte می‌باشد.</p>	
<p>هیچ مقدار بازگشتی ندارد</p>	

SetDataBlockStr	نوشتن اطلاعات در حافظه بصورت رشته ای
<pre>procedure SetDataBlockStr(const sVID:WideString; const sPWD: WideString; Start: Word;DataLen: Word; const DataBlockStr:WideString;const defChar: WideString=' ');</pre>	
<p>در صورتیکه بخواهید بیش از یک بایت را بصورت رشته ای در حافظه ذخیره نمائید از این روتین استفاده کنید.</p>	
<p>sVID: این پارامتر VID می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و باید بصورت WideString باشد.</p> <p>sPWD: این پارامتر رمز عبور قفل می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد.</p> <p>Start: آدرس شروع برای ذخیره اطلاعات و از نوع Word</p> <p>DataLen: این متغیر که از نوع Word می‌باشد اندازه بلوک مورد نظر برای نوشتن را مشخص می‌نماید.</p> <p>DataBlockStr: اطلاعات مورد نظر در این متغیر که یک آرایه از نوع WideString می‌باشد ذخیره و پس از اجرای روتین در قفل نوشته می‌شود.</p> <p>defChar: در صورتیکه رشته مورد نظر کمتر از طول ذکر شده باشد بقیه حافظه با این کاراکتر پر می‌شود .</p>	
<p>هیچ مقدار بازگشتی ندارد</p>	



SetDataHexBlock	نوشتن بلوک HEX در بخش داده
<pre>procedure SetDataHexBlock(const sVID: WideString; const sPWD: WideString; Start: Word; DataLen: Word; const DataHexBlock: WideString);</pre>	
<p>در صورتیکه بخواهید بیش از یک بایت را بصورت رشته HEX در حافظه ذخیره نمائید باید از این روتین استفاده نمائید.</p>	
<p>sVID: پارامتر VID می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و باید بصورت WideString باشد.</p> <p>sPWD: این پارامتر رمز عبور قفل می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد.</p> <p>Start: آدرس شروع برای ذخیره اطلاعات و از نوع Word</p> <p>DataLen: این متغیر که از نوع Word می‌باشد اندازه بلوک مورد نظر برای نوشتن را مشخص می‌نماید.</p> <p>DataHexBlock: اطلاعات مورد نظر در این متغیر که یک آرایه از نوع WideString می‌باشد ذخیره و پس از اجرای روتین در قفل نوشته می‌شود.</p>	
<p>هیچ مقدار بازگشتی ندارد</p>	

GetDataByte	خواندن یک بایت از حافظه قفل
<pre>function GetDataByte(const sVID: WideString; const sPWD: WideString; Address: Word): Byte;</pre>	
<p>برای خواندن اطلاعات از قفل بصورت بایت به بایت از این روتین استفاده می‌شود این روتین از آدرس مشخص شده اطلاعات یک بایت را می‌خواند.</p>	
<p>sVID: پارامتر VID می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و باید بصورت WideString باشد.</p> <p>sPWD: این پارامتر رمز عبور قفل می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString می‌باشد.</p> <p>Address: آدرس مورد نظر جهت خواندن اطلاعات و از نوع Word می‌باشد.</p>	
<p>مقدار برگشتی : در صورت اجرای موفقیت مقدار موجود در آدرس مورد نظر برگشت داده می‌شود و نوع آن byte می‌باشد.</p>	



GetDataBlockStr	خواندن اطلاعات از حافظه بصورت رشته ای
<pre>function GetDataBlockStr(const sVID: WideString; const sPWD: WideString; Start: Word; DataLen: Word): WideString;</pre>	
<p>در صورتیکه بخواهید بیش از یک بایت را بصورت رشته ای از حافظه بخوانید باید از این روتین استفاده نمائید.</p>	
<p>sVID: پارامتر VID می باشد که قبل از اجرای روتین باید مقدار دهی شود و باید بصورت WideString باشد.</p> <p>sPWD: این پارامتر رمز عبور قفل می باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد.</p> <p>Start: آدرس شروع برای خواندن اطلاعات و از نوع Word</p> <p>DataLen: این متغیر که از نوع Word می باشد اندازه بلوک مورد نظر برای خواندن را مشخص می نماید.</p>	
<p>مقدار برگشتی : اطلاعات مورد نظر پس از اجرای موفقیت آمیز روتین بصورت یک widestring برگشت داده می شود.</p>	

GetDataHexBlock	خواندن بلوک HEX از بخش داده
<pre>function GetDataHexBlock(const sVID: WideString; const sPWD: WideString; Start: Word; DataLen: Word): WideString;</pre>	
<p>در صورتیکه بخواهید بیش از یک بایت را بصورت رشته HEX از حافظه بخوانید از این روتین استفاده نمائید.</p>	
<p>sVID: این پارامتر VID می باشد که قبل از اجرای روتین باید مقدار دهی شود و باید بصورت WideString باشد.</p> <p>sPWD: این پارامتر رمز عبور قفل می باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد.</p> <p>Start: آدرس شروع برای خواندن اطلاعات و از نوع Word</p> <p>DataLen: این متغیر که از نوع Word می باشد اندازه بلوک مورد نظر برای خواندن را مشخص می نماید.</p>	
<p>مقدار برگشتی : اطلاعات مورد نظر پس از اجرای موفقیت آمیز روتین بصورت یک widestring برگشت داده می شود.</p>	



GetEncryption	رمزنگاری
<pre>function GetEncryption(const sVID: WideString; const sPWD: WideString; const sPData: WideString; Repetition: Word): WideString;</pre>	
<p>یکی از روتین های مهم و کاربردی در قفل عمل رمزنگاری (Encryption) می باشد. این روتین با استفاده از کلید ذخیره شده در خود قفل ، اطلاعات ارسالی را رمزنگاری می نماید ، این عمل می تواند بصورت متوالی انجام شود به این صورت که پارامتر ارسالی بعد از رمزنگاری شدن ، دوباره رمزنگاری شود.</p>	
<p>sVID: این پارامتر VID می باشد که برای اجرای روتین باید مقدار دهی شود و بصورت WideString می باشد. sPWD: این پارامتر رمز عبور قفل می باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد. sPData: اطلاعات مورد نظر برای رمزنگاری بصورت WideString در این پارامتر قرار می گیرد. Repetition: این متغیر که از نوع Word می باشد تعداد دفعات برای انجام عمل رمزنگاری را مشخص می کند این عدد حداقل باید یک باشد.</p>	
<p>مقدار برگشتی : اطلاعات رمزنگاری شده بعد از اجرای موفقیت آمیز روتین و از نوع WideString برگشت داده می شود.</p>	

GetDecryption	رمزگشایی
<pre>function GetDecryption(const sVID: WideString; const sPWD: WideString; const sCData: WideString; Repetition: Word): WideString;</pre>	
<p>برای رمزگشایی اطلاعات رمزنگاری شده از این روتین استفاده می شود. به این صورت که در صورتیکه عددی را رمزنگاری و سپس عدد بدست آمده را رمزگشایی نمائیم ، عدد اول بدست می آید به این شرط که تعداد تکرار در هر دو مرحله یکسان باشد.</p>	
<p>sVID: این پارامتر VID می باشد که برای اجرای روتین باید مقدار دهی شود و بصورت WideString می باشد. sPWD: این پارامتر رمز عبور قفل می باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد. sCData: اطلاعات رمزنگاری شده که می خواهیم رمزگشایی شود و به صورت WideString می باشد. Repetition: این متغیر که از نوع Word می باشد تعداد دفعات برای انجام عمل رمزگشایی را مشخص می کند.</p>	
<p>مقدار برگشتی : در صورت اجرای موفقیت آمیز : اطلاعات رمزگشایی شده برگشت داده می شود که از نوع WideString می باشد.</p>	



GetNET	دریافت حداکثر تعداد کاربران مجاز قفل شبکه
function GetNET: Word;	
با توجه تنظیمات شبکه این تابع تعداد حداکثر مجاز کاربران استفاده کننده از قفل تحت شبکه را نشان می‌دهد. در صورت ست نشدن برای شبکه خطای ۶ (Code not set) برمی‌گرداند	
هیچ پارامتری ندارد	
مقدار برگشتی: از نوع BYTE می‌باشد و بعد از اجرا مقدار مورد نظر را بر می‌گرداند. در صورت ست نشدن برای شبکه خطای ۶ (Code not set) اتفاق می‌افتد	

GetPID	دریافت PID
function GetPid(const sVID: WideString; const sPWD: WideString): WideString;	
PID ذخیره شده در قفل توسط این سرویس قابل دسترسی می‌باشد. PID با استفاده از نرم افزار برنامه ریز در قفل ذخیره می‌گردد	
sVID: این پارامتر VID می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و باید بصورت WideString باشد.	
sPWD: این پارامتر رمز عبور قفل می‌باشد که قبل از اجرای روتین باید مقدار دهی شود و بصورت WideString باشد.	
مقدار برگشتی: در صورت اجرای موفقیت آمیز مقدار PID به صورت Widestring برگشت داده می‌شود.	

ConvDelimitedStringToString	تبدیل رشته از DNString به رشته کاراکتری
function ConvDelimitedStringToString(const strDCnv: WideString;const ALen: Word=16;const Sep: WideString='.'): WideString;	
با استفاده از این روتین می‌توان رشته کاراکتری را به DN تبدیل کرد	
strDCnv: این پارامتر رشته ورودی می‌باشد که می‌خواهیم تبدیل کنیم و به صورت WideString می‌باشد.	
ALen: طول رشته ای که قرار است تبدیل شود را مشخص می‌کند.	
Sep: کاراکتر واسط در خروجی را تعیین می‌کند که به صورت پیشفرض '.' است	
مقدار برگشتی: در صورت اجرای موفقیت آمیز رشته تبدیل شده بصورت کاراکتری و WideString برگشت داده می‌شود.	



ConvStringToDelimitedString	تبدیل رشته از رشته کاراکتری به DNString
<pre>function ConvStringToDelimitedString(const strCnv: WideString;const ALen: Word=16;const Sep: WideString='.'): WideString;</pre>	
<p>با استفاده از این روتین می‌توان رشته کاراکتری را به DN تبدیل کرد</p>	
<p>strCnv: این پارامتر رشته ورودی می‌باشد که می‌خواهیم تبدیل کنیم و رشته ای از نوع WideString باشد. Sep: کاراکتر واسط در خروجی را تعیین می‌کند که به صورت پیشفرض '.' است</p>	
<p>مقدار برگشتی: در صورت اجرای موفقیت آمیز رشته تبدیل شده بصورت کاراکتری و WideString برگشت داده می‌شود.</p>	

ConvHexStringToDelimitedString	تبدیل رشته HEX به DNString
<pre>function ConvHexStringToDelimitedString(strHCnv: WideString;const ALen: Word=16;const Sep: WideString='.'): WideString;</pre>	
<p>با استفاده از این روتین می‌توان رشته مبنای ۱۶ (HEX) را به DN تبدیل کرد</p>	
<p>strHCnv: این پارامتر رشته ورودی می‌باشد که می‌خواهیم تبدیل کنیم و به صورت WideString می‌باشد. ALen: طول رشته ای که قرار است تبدیل شود را مشخص می‌کند. Sep: کاراکتر واسط در خروجی را تعیین می‌کند که به صورت پیشفرض '.' است</p>	
<p>مقدار برگشتی: در صورت اجرا رشته تبدیل شده بصورت کاراکتری و WideString برگشت داده می‌شود.</p>	

ConvDelimitedStringToHexString	تبدیل DNString به رشته HEX
<pre>function ConvDelimitedStringToHexString(const strDCnv: WideString;const ALen: Word=16;const DSep: WideString='.';const HSep: WideString=' '): WideString;</pre>	
<p>با استفاده از این روتین می‌توان رشته DN را به رشته مبنای ۱۶ (HEX) تبدیل کرد</p>	
<p>strDCnv: این پارامتر رشته ورودی می‌باشد که می‌خواهیم تبدیل کنیم و به صورت WideString می‌باشد. ALen: طول رشته ای که قرار است تبدیل شود را مشخص می‌کند. Dsep و Hsep: کاراکتر واسط در ورودی و خروجی را تعیین می‌کند که به صورت پیشفرض '.' و ' ' است</p>	
<p>مقدار برگشتی: در صورت اجرا رشته تبدیل شده بصورت کاراکتری و WideString برگشت داده می‌شود.</p>	



ConvStringToHexString	تبدیل رشته معمولی به رشته HEX
<pre>function ConvStringToHexString(const strCnv: WideString;const ALen: Word=16;const Sep: WideString=' '): WideString;</pre>	
<p>با استفاده از این روتین می‌توان رشته معمولی را به رشته مبنای ۱۶ (HEX) تبدیل کرد</p>	
<p>strCnv: این پارامتر رشته ورودی می‌باشد که می‌خواهیم تبدیل کنیم و به صورت WideString می‌باشد.</p> <p>ALen: طول رشته ای که قرار است تبدیل شود را مشخص می‌کند.</p> <p>HSep: کاراکتر واسط در خروجی را تعیین می‌کند که به صورت پیش‌فرض ' ' است</p>	
<p>مقدار برگشتی : در صورت اجرای موفقیت آمیز رشته تبدیل شده بصورت کاراکتری و WideString برگشت داده می‌شود.</p>	

ConvHexStringToString	تبدیل رشته HEX به رشته معمولی
<pre>function ConvHexStringToString(strHCnv: WideString;const ALen: Word=16): WideString;</pre>	
<p>با استفاده از این روتین می‌توان رشته مبنای ۱۶ (HEX) را به رشته معمولی تبدیل کرد</p>	
<p>strHCnv: این پارامتر رشته ورودی می‌باشد که می‌خواهیم تبدیل کنیم و به صورت WideString می‌باشد.</p> <p>ALen: طول رشته ای که قرار است تبدیل شود را مشخص می‌کند.</p>	
<p>مقدار برگشتی : در صورت اجرای موفقیت آمیز رشته تبدیل شده بصورت کاراکتری و WideString برگشت داده می‌شود.</p>	

GetTimerCounter	دریافت مقدار شمارنده/زمانسنج
<pre>Function GetTimerCounter: integer;</pre>	
<p>در صورتی که تنظیمات خاصی برای محدود کردن دفعات یا مدت زمان استفاده از قفل تعیین شده باشد با استفاده از این روتین می‌توان از آن تنظیمات مطلع شد.</p>	
<p>هیچ پارامتری ندارد</p>	
<p>مقدار بازگشتی مقدار عدد موجود در پارامتر شمارنده قفل است. در صورتی که کاهنده خودکار (Timer) ست نشده باشد خطای Nlevt_TimerNS در پارامتر خطا مشاهده خواهد شد.</p>	



Activate	فعالسازی مجدد قفل
procedure Activate(ActivationCode: WideString);	
در صورتی که در طول استفاده از قفل مقدار شمارنده به صفر برسد، قفل به حالت Suspend می‌رود که برای خروج از این حالت و غیر فعال کردن شمارنده/زمانسنج می‌توان با استفاده از یک کد فعال سازی صحیح قفل را از طریق این روتین فعال کرد.	
ActivationCode : رشته فعالسازی است که توسط NLBuilder ایجاد می‌گردد.	
هیچ مقدار بازگشتی ندارد	

DecreaseCounter	کاهش یک واحد از شمارنده
procedure DecreaseCounter;	
در طول استفاده از قفل می‌توان مقداری را به عنوان دفعات استفاده از قفل تعیین و با استفاده از این روتین در مقاطع مورد نظر آنرا یک واحد کاهش داد.	
هیچ پارامتری ندارد	
هیچ مقدار بازگشتی ندارد	

ErrNo	کد خطای حاصل از آخرین متد اجرا شده
property ErrNo: Word	
توابع فوق در صورت اجرای موفق باعث بازگشت مقدار ۰-۰- از طریق این property می‌شوند ولی در صورت بروز خطا این پارامتر عدد غیر صفر بر می‌گرداند.	
این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع word بر می‌گرداند	

ErrDescr	پیام خطای متناسب با کد خطا (انگلیسی)
property ErrDescr: WideString	
توابع در صورت اجرای موفق باعث بازگشت کلمه 'OK' از طریق این property می‌شوند ولی در صورت بروز خطا پیام متناسب با نوع خطا از این مشخصه قابل دریافت است.	



این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع **widestring** بر می‌گرداند

ErrDescrFA

پیام خطای متناسب با کد خطا (فارسی)

property ErrDescrFA: WideString

توابع در صورت اجرای موفق باعث بازگشت کلمه «تایید» از طریق این **property** می‌شوند ولی در صورت بروز خطا پیام متناسب با نوع خطا از این مشخصه قابل دریافت است.

این مشخصه به طور یکطرفه خواندنی است و مقدار عددی از نوع **widestring** بر می‌گرداند



پیوست ۱: کد خطاها

در جدول زیر تمام خطاهای ممکن در استفاده از کتابخانه رابط قفل در برنامه نویسی آمده است:

توضیح	سمبل خطا	شماره خطا
بدون خطا	nErr_OK	۰
قفل پیدا نشد	nErr_DNF	۱
پارامتر ورودی غیرمعتبر است	nErr_INP	۲
خطای IO	nErr_IOE	۳
پارامتر ورودی اشتباه است	nErr_WRP	۴
کد مورد نظر ست شده است	nErr_CS	۵
کد مورد نظر ست نشده است	nErr_CNS	۶
قفل مشغول است	nErr_BSY	۸
نگارش قفل ناسازگار است	nErr_ICV	۱۰
دسترسی غیر مجاز است	nErr_ACD	۱۱
VID یا رمز عبور اشتباه است	nErr_IVP	۱۲
قفل غیرفعال شده است	nErr_DIS	۱۳
قفل در مسدود است	nErr_BLK	۱۴
قفل به حالت تعلیق در آمده است	nErr_SUS	۱۵
خطای ناشناخته	nErr_OTE	۱۶
سرویس مورد نظر پشتیبانی نمی شود	nErr_NSD	۱۷
خطا در HID (رابط USB)	nErr_HID	۱۸
قفل غیرفعال فجیع شده است	nErr_CAT	۱۹



سرور شبکه قفل اشغال است	nErr_NTBSY	۲۰
اتصال به سرور قبول شد	nErr_NTAPT	۲۱
اتصال به سرور رد شد	nErr_NTRJT	۲۲
سرور شبکه در دسترس نیست	nErr_NTNAV	۲۳
ارتباط با سرور قطع شده است	nErr_NTDC	۲۴
ارتباط با سرور برقرار است	nErr_NTCD	۲۵
طول رشته بیش از حد مجاز است	nErr_SBig	۲۶
خاصیت شبکه قفل غیرفعال است	nErr_NETNS	۵۰
کلید رمزنگاری AES تنظیم نشده است	nErr_KEYNS	۵۱
کاهنده خودکار (زمانسنج ۵,۵۹ ثانیه ای) فعال نیست	nErr_TimerNS	۵۶
سرویس غیرفعال است احتمالاً دستگاه غیرفعال یا مسدود است	nErr_SCD	۱۲۹
هیچ دستگاهی انتخاب نشده است	nErr_DNS	۱۳۲



پیوست ۲: توصیه های امنیتی

بدین منظور می‌توانید اطلاعات و پارامترهایی را که توسط نرم افزار برنامه ریز روی قفل خام تنظیم نموده اید توسط متدهای موجود در کتابخانه رابط مورد بررسی قرار دهید. برای بالا بردن ضریب امنیتی نرم افزارهای نوشته شده می‌توانید موارد زیر را رعایت کنید. حتما از رمز عبور استفاده کنید. رمز عبور را در نرم افزار برنامه ریز باید تنظیم نمائید و حتی المقدور رمز عبور را ۱۶ کاراکتر وارد نمائید.

کلید مربوط به کد گذاری AES را حتما تنظیم نمائید و در سورس نرم افزارهایتان حتما از روتین های Encrypt و Decrypt استفاده نمائید. ساده ترین روش برای این منظور این است که عددی را با این الگوریتم Encrypt نمائید. سپس عدد بدست آمده را در برنامه با روتین Decrypt رمز گشایی نمائید اگر عدد برگشتی با عدد اول برابر بود یعنی قفل موجود است، همچنین می‌توانید به تعداد نامحدود عدد های مختلف را به قفل ارسال و کد گذاری شده آن را بدست آورید (نرم افزار برنامه ریز این کار را برای شما انجام می‌دهد) به این صورت که شما تعداد عدد مورد نظر را وارد و سپس به تعداد مشخص شده عدد ایجاد و به قفل ارسال می‌شود این عدد و عدد کدگذاری شده آن در یک فایل متنی ذخیره می‌شوند و حال شما می‌توانید این اعداد را در بین اطلاعات خود مخفی نمائید. به طور مثال در بانک های خود فیلدهائی اضافه و این اعداد را آنجا قرار دهید و یا می‌توانید کلیه اعداد را در یک فایل باینری با فرمتی که خودتان انتخاب می‌کنید قرار دهید و در نرم افزار به طور تصادفی عددها را خوانده به قفل ارسال و جواب آن را با جواب موجود در بانک خود مقایسه کنید، در صورت برابر بودن مطمئن باشید که قفل موجود می‌باشد. علاوه بر این کلیدی را که داخل قفل ست کرده اید را می‌توانید به صورت گسسته لابلای اطلاعات خود پنهان کرده و در هنگام تست قفل اطلاعات تصادفی را هم با قفل و هم با کلید داخل برنامه به روش Rijndael ECB Mode (AES) رمزنگاری کرده و نتایج را مقایسه نمائید.

از دیگر روش ها برای بالا بردن ضریب امنیت، استفاده از نرم افزارهای فشرده سازی برای فشرده کردن فایل اجرایی برنامه می‌باشد. این برنامه ها اطلاعات اضافی در فایل اجرایی را حذف می‌کنند و مزیت آن اینست که اولاً حجم فایل کم می‌شود، ثانياً فایل اجرایی غیر قابل Trace می‌شود.

ضمناً برای جلوگیری از دستکاری فایل اجرایی خود طول فایل را در هنگام اجرا چک کنید. همچنین برای شناسایی کتابخانه تقلبی LNovinLock.dll که یکی از راههای دور زدن سیستم می‌تواند باشد با استفاده از کلاس CapiCom در ویندوز، امضای دیجیتال فایل مورد نظر را چک کنید که حتماً متعلق به شرکت نوین افزار باشد.

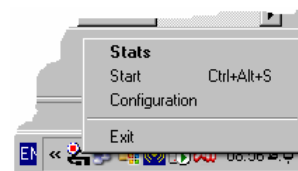


پیوست ۳: قفل سخت افزاری تحت شبکه

بدین منظور کفایت نرم افزار اشتراک گزارنده قفل در شبکه را روی سیستم خاصی فعال کرد و پس از اطلاع از IP یا اسم کامپیوتر سرور و پورتی که در نرم افزار رابط تنظیم می‌گردد با استفاده از کلاس clsNetClient از دیگر کامپیوتر های از طریق شبکه به این نرم افزار وصل شد. پس از وصل شدن دقیقاً مانند یک قفل محلی می‌توان با آن رفتار کرد. در مورد کلاس فوق در بخشهای قبلی توضیح دادیم. در ادامه بحث به توضیح در مورد نرم افزار سرور شبکه قفل می‌پردازیم. این نرم افزار از دو قسمت تشکیل شده، یکی سرویس که در پشت صحنه ویندوز در حال اجرا است و دومی نماینده^۱ ای که وظیفه کنترل سرویس را به عهده دارد.

NovinLockNetAgent: نماینده این سرویس بعد از ورود به ویندوز به

صورت یک نشانه در کنار ساعت به صورت Tray نمایش داده می‌شود. این نماینده علاوه بر کنترل سرویس از طریق منوی اصلی خود می‌تواند از طریق پارامترهای خط



فرمان نیز سرویس را کنترل کند:

/start: برای استارت سرویس مربوطه

/stop: برای ایست سرویس

/restart: برای راه اندازی مجدد سرویس

NovinLockNetServerSVC: این نرم افزار می‌زبان کاربران شبکه برای استفاده از قفل شبکه

می‌باشد:

این نرم افزار به صورت یک سرویس در ویندوز اضافه شده و استفاده از آن لزوماً نیازی به ورود

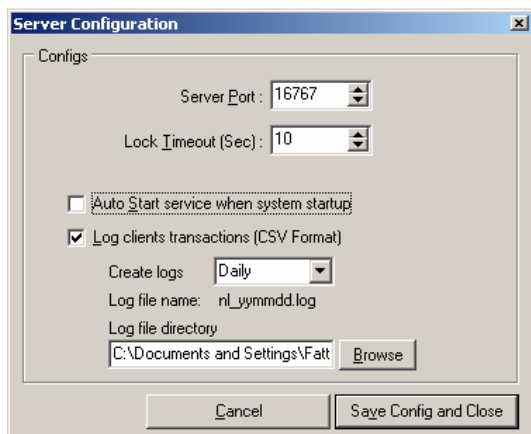
(Login) به ویندوز ندارد.

برای تغییر تنظیمات این سرویس روی این نشان نماینده راست کلیک کنید تا منوی مربوطه نشان داده

شود و مورد Configuration را انتخاب نمایید.

در فرم مربوطه ۶ مورد را می‌توان تنظیم کرد، این موارد در registry ویندوز ثبت شده و در دفعات

بعدی اجرا نیازی به تنظیم مجدد آنها نیست :



۱. Server port: پورت سرویس دهی به

کاربران که می‌تواند عددی بین ۱ تا ۶۵۵۳۵



باشد. ولی توصیه می‌شود جهت عدم اختلال در کار سایر نرم افزار از پورتهای استاندارد مانند ۲۱، ۸۰، ۱۳۵، ۱۳۹، ۴۴۵، ۱۴۳۳، ۳۳۸۹ و ... استفاده نشود. همچنین در صورت استفاده از نرم افزارهای محافظ مانند Firewall ها استفاده از آن پورت را مجاز نمایید.

۲. Lock Timeout: در صورتی که هر یک از Client ها پیام اعلان موجودیت به سرور ارسال نکنند بعد از این مدت سرور آنها را disconnect می‌کند.

۳. Auto Start ...: باعث می‌شود که در هنگام بارگزاری ویندوز این سرویس به طور خودکار فعال شده و آماده بهره برداری توسط کاربران شبکه شود.

۴. Log clients ...: در صورت فعالسازی این مورد، نرم افزار از فعالیتهای کاربران روی قفل گزارش متنی تهیه می‌کند.

۵. Create logs: تعیین می‌کند که نوع گزارشگیری چگونه باشد. به طور مثال در صورت انتخاب Hourly در هر ساعت یک فایل ایجاد می‌شود و گزارشها در داخل آن ریخته می‌شود.

۶. Log file directory: تعیین می‌کند که فایل‌های گزارشگیری در کجای سیستم ذخیره شود. ساختار این گزارشها به صورت زیر است :

#Date: 2006/05/14 21:20:37 → تاریخ و ساعت شروع گزارشگیری

#Fields: time c-ip c-port cs-command sc-result sc-bytes cs-bytes

فیلدهای اطلاعاتی به ترتیب از چپ عبارتند از :

زمان انجام / آدرس IP کاربر / پورت کامپیوتر کاربر / متدی که اجرا شده / کد نتیجه / مقدار بایت ارسالی از سرور / مقدار بایت دریافتی از کلاینت

گزارش آماری از کاربران شبکه متصل شده:

با دوبار کلیک روی نشانه نرم افزار یا انتخاب Stats از منوی نرم افزار پنجره زیر باز می‌شود که نشان

دهنده اطلاعاتی در مورد وضعیت نرم افزار، قفل شبکه و کاربرانی که از آن استفاده می‌کنند را در اختیار ما قرار می‌دهد.



Server Statistic					
Startup at : 2007/01/13 - 12:55:22 È.Ù		Lock Serial : --			
Capacity: --		Lock Version : --			
Client Count : 0		Memory Size : --			
Code	Client IP	Client Port	Connection Duration	Idle Time	Transactions
0					
<div style="text-align: right;"> <input type="button" value="Configuration"/> <input type="button" value="Disconnect Selected"/> <input type="button" value="Exit Stat"/> </div>					

نکته ای که قابل توجه است که متد Start در clsNetClient زمانی درست انجام می‌شود که علاوه بر منطبق بودن Host و Port تعیین شده در آن با تنظیمات نرم افزار سرور باید قفلی که روی سرور قرار دارد برای شبکه ست شده باشد و همچنین ظرفیت تعداد کاربران آن نیز پر نشده باشد. در غیر این صورت clsNetClient خطای Connection rejected برمی گرداند.

برای درک بهتر موضوع به مثال این کلاس واقع در شاخه نمونه کدهایی که در شاخه Net Client هر زبان برنامه نویسی روی CD همراه قفل بهره ببرید.