SuperTest – ou comment TrustInSoft garantit un code source 100 % exempt de bogues à ses clients

La société d'outils de développement logiciel TrustInSoft, qui dessert les secteurs internationaux de l'aéronautique, des télécommunications, de l'IoT industriel et de l'automobile via ses bureaux de Paris (France) et de San Francisco (CA, États-Unis), n'est pas un client typique de la suite de test et de validation de compilateur SuperTest de Solid Sands. Le produit phare de la société - TrustInSoft Analyzer - est un analyseur de code source avancé qui vérifie l'exactitude d'un programme C ou C++ avant qu'il ne s'approche d'un compilateur. Alors, pour une entreprise qui n'est pas particulièrement intéressée par le développement ou la validation de compilateurs, pourquoi TrustInSoft utilise-t-elle SuperTest ? La réponse réside dans le besoin de certification de la sécurité fonctionnelle ISO 26262 d'un bout à l'autre de la chaîne d'outils - du code source au binaire.

L'analyseur de code source de TrustInSoft crée un modèle mathématiquement rigoureux d'un programme C ou C++ qui, associé à des informations sur les paramètres d'entrée du programme et l'architecture cible, permet aux utilisateurs de vérifier avec une précision de 100 % que leur programme ne contient pas de comportements non définis et qu'il est conforme à ses exigences fonctionnelles.

« En utilisant des méthodes formelles, nous pouvons fournir une réponse mathématiquement définitive sur le comportement d'un programme C ou C++ - une garantie mathématique de l'absence de bogues. Ainsi, si notre TrustInSoft Analyzer dit qu'un programme est correct, vous ne trouverez aucune violation dans ce programme, quels que soient les vecteurs d'entrée que vous utilisez », déclare Benjamin Monate, cofondateur et directeur général de TrustInSoft Analyzer, à l'occasion d'une conférence de presse.

En ce qui concerne la compilation, TrustInSoft Analyzer ne demande que des informations minimales sur le compilateur, par exemple s'il s'agit d'une cible de type GCC et s'il compile pour une architecture 32 ou 64 bits.

« Pour l'analyse, nous ne compilons jamais rien. Ainsi, bien que certaines informations relatives au compilateur soient nécessaires pour paramétrer l'analyseur, nous supposons que lorsque le compilateur est finalement invoqué, il fonctionnera comme prévu. C'est donc au client de choisir un compilateur qui implémente correctement la norme C ou C++ »,

explique Benjamin Monate. « Bien entendu, il doit s'assurer que son compilateur fonctionne correctement à l'aide d'une suite de tests telle que SuperTest, car si le compilateur traduit incorrectement son code source, il n'est pas garanti que les propriétés que nous prouvons mathématiquement sur le code source soient transférées à l'exécutable. Notez que même si un compilateur est parfaitement conforme aux normes, la compilation d'un code qui contient un comportement indéfini générera un exécutable qui n'a aucun sens et qui peut contenir des problèmes de sûreté et de sécurité graves et imprévisibles. C'est pourquoi vous devez utiliser TrustInSoft Analyzer et SuperTest ».
Pour garantir que le code source et l'exécutable ont la même sémantique, le plus important est que l'analyseur et le compilateur interprètent le langage C ou C++ précisément comme défini dans la norme correspondante. Pour TrustInSoft, SuperTest est l'un des outils utilisés pour vérifier la conformité de TrustInSoft Analyzer.
« Parce que notre outil fournit une solution qui offre des garanties mathématiques, nous devons être extrêmement stricts sur la façon dont nous développons l'analyseur lui-même, ce qui signifie que nous devons le valider rigoureusement. Et pour le marché de l'automobile, nous avions particulièrement besoin d'une qualification ISO 26262. Nous disposons déjà de notre propre gamme étendue de suites de tests, mais un élément important pour fournir les preuves nécessaires à la norme ISO 26262 était d'utiliser une suite de tests indépendante qui est déjà largement reconnue dans l'industrie électronique automobile. C'est pourquoi nous avons choisi SuperTest », explique Benjamin Monate.

La société utilise SuperTest pour confirmer que TrustInSoft Analyzer interprète correctement la sémantique des langages C et C++.
« Grâce à SuperTest, nous avons identifié quelques parties subtiles de la norme qui n'étaient pas encore totalement prises en charge par TrustInSoft Analyzer et qui sont désormais implémentées et qualifiées. Nous avons également pu identifier quelques divergences mineures dans la manière dont Solid Sands et nos propres ingénieurs interprétaient les normes C et C++, ce qui a rendu cette collaboration très enrichissante, et nous espérons la poursuivre », déclare Benjamin Monate. « Ce qui compte vraiment pour nous, c'est que SuperTest nous a aidés à extraire une matrice de couverture de la norme et à la présenter comme une preuve de la qualité de notre outil - que nous ne disons jamais quelque chose d'incorrect à propos du code - sur la base d'une suite de tests complètement indépendante qui est soigneusement conçue pour vérifier tous les coins possibles. »

Pour les développeurs de logiciels de systèmes de sécurité critiques qui ont besoin d'une qualification ISO 26262 de bout en bout de leur chaîne d'outils, l'utilisation d'une combinaison de TrustInSoft Analyzer et SuperTest pour valider leur code source et assurer une compilation correcte est une solution puissante qui s'intègre parfaitement dans les environnements d'intégration continue.

www.trust-in-soft.com
www.solidsands.nl